



UNIVERSIDAD CARLOS III DE MADRID

ESCUELA POLITÉCNICA SUPERIOR

INGENIERÍA TÉCNICA INFORMÁTICA DE GESTIÓN

PROYECTO FIN DE CARRERA

# **“Herramienta de soporte a la generación de planificaciones de proyectos enriquecidas con Patrones de Producto”**

Autor: David Gómez Pedrero  
Tutores: Fuensanta Medina Domínguez  
Fecha: Diciembre, 2009



Mi infinito agradecimiento a M.G.P,  
quien estuvo en el momento y lugar adecuados,  
a pesar de la distancia.  
Un viaje entre los sueños  
y los recuerdos.

Tvb.



Dedicado a todos los que ya no están,  
especialmente a Ron.  
así como a los familiares, amigos y viajeros,  
por su apoyo, paciencia y consejos.

Ast.



## ÍNDICE DE CONTENIDOS

1	INTRODUCCIÓN .....	14
1.1	DESCRIPCIÓN GENERAL DEL PROYECTO .....	14
1.2	EL PROCESO DE GESTIÓN DE PROYECTOS .....	17
1.2.1	Ubicación de la Gestión de Proyectos en el proceso de la construcción del Software.....	17
1.2.2	Procesos/Actividades de la Gestión de Proyectos.....	18
1.3	EL PROCESO DE PLANIFICACIÓN .....	26
1.4	MOTIVACIÓN.....	27
1.5	METODOLOGÍAS Y TÉCNICAS PARA EL PROCESO DE PLANIFICACIÓN.....	28
1.6	HERRAMIENTAS EXISTENTES PARA EL PROCESO DE PLANIFICACIÓN.....	36
1.7	OBJETIVOS .....	42
2	PLANIFICACIÓN Y ESPECIFICACIÓN DE REQUISITOS SOFTWARE .....	44
2.1	INTRODUCCIÓN.....	44
2.1.1	Propósito .....	44
2.1.1.1	Audiencia a la que va dirigido.....	45
2.1.2	Alcance.....	45
2.1.2.1	Funcionalidades.....	46
2.1.2.2	Beneficios, objetivos y metas.....	47
2.2	DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS .....	47
2.2.1	Definiciones .....	47
2.2.2	Abreviaturas .....	50
2.2.1	Acrónimos .....	51
2.2.2	Apreciación Global .....	52
2.3	DESCRIPCIÓN GLOBAL.....	52
2.3.1	Perspectiva del Producto.....	52
2.3.2	Funciones del Producto .....	53
2.3.3	Requisitos Funcionales.....	55
2.3.4	Requisitos de Interfaz.....	57
2.3.4.1	Interfaz con el Usuario. ....	57

2.3.5	Requisitos No Funcionales.....	57
2.3.5.1	Requisitos Hardware.....	57
2.3.5.2	Requisitos Software.....	58
<b>2.4</b>	<b>CASOS DE USO.....</b>	<b>59</b>
2.4.1	Definición de los Casos de Uso.....	59
2.4.2	Diagrama de casos de uso.....	59
2.4.3	Casos de Uso de Alto Nivel.....	61
2.4.3.1	Entrada al sistema.....	61
2.4.3.2	Selección de Parámetros de Búsqueda.....	61
2.4.3.3	Asignación de Patrones.....	62
2.4.3.4	Actualización de Tareas.....	63
2.4.4	Casos de Uso en Formato Expandido.....	64
2.4.4.1	Caso de Uso: Autenticación.....	64
2.4.4.2	Caso de Uso: Probar Descriptor WSDL.....	66
2.4.4.3	Caso de Uso: Seleccionar Parámetros.....	67
2.4.4.4	Caso de Uso: Realizar Búsqueda de Patrones.....	69
2.4.4.5	Caso de Uso: Seleccionar Patrón.....	70
2.4.4.6	Caso de Uso: Consultar Información del Patrón.....	71
2.4.4.7	Caso de Uso: Seleccionar Tareas de Destino.....	73
2.4.4.8	Caso de Uso: Crear Columna Destino.....	74
2.4.4.9	Caso de Uso: Guardar proyecto en BBDD.....	75
<b>2.5</b>	<b>PLANIFICACIÓN.....</b>	<b>77</b>
<b>3</b>	<b>ANÁLISIS.....</b>	<b>80</b>
<b>3.1</b>	<b>MODELO CONCEPTUAL.....</b>	<b>80</b>
<b>4</b>	<b>DISEÑO.....</b>	<b>81</b>
<b>4.1</b>	<b>DIAGRAMA DE CLASES DE DISEÑO.....</b>	<b>83</b>
<b>4.2</b>	<b>DIAGRAMAS DE SECUENCIA.....</b>	<b>84</b>
4.2.1	Diagrama de Secuencia: CU Autenticación.....	84
4.2.2	Diagrama de Secuencia: CU Probar Descriptor WSDL.....	85
4.2.3	Diagrama de Secuencia: CU Seleccionar Parámetros.....	85
4.2.4	Diagrama de Secuencia: CU Realizar Búsqueda de Patrones.....	86
4.2.5	Diagrama de Secuencia: CU Seleccionar Patrón.....	86
4.2.6	Diagrama de Secuencia: CU Consultar Información del Patrón.....	87
4.2.7	Diagrama de Secuencia: CU Seleccionar Tareas Destino.....	87
4.2.8	Diagrama de Secuencia: CU Crear Columna Destino.....	88
4.2.9	Diagrama de Secuencia: CU Guardar proyecto en BBDD.....	88
<b>4.3</b>	<b>ENTORNO DEL DISEÑO – ESQUEMA DE COMUNICACIÓN.....</b>	<b>89</b>
<b>4.4</b>	<b>CLIENTE DE LOS SERVICIOS WEB.....</b>	<b>90</b>



4.5	SERVIDOR DE LOS SERVICIOS WEB .....	92
4.6	DIAGRAMA DE FLUJO .....	93
4.7	MODELO DE DATOS .....	96
5	IMPLEMENTACIÓN .....	99
6	CONCLUSIONES Y TRABAJO FUTURO .....	108
6.1	CONCLUSIONES .....	108
6.2	TRABAJO FUTURO .....	110
7	BIBLIOGRAFÍA .....	112
7.1	MANUALES .....	112
7.2	ENLACES WEB.....	114

## **ANEXOS**

<i>ANEXO A. MANUAL DE INSTALACIÓN</i> .....	118
1 <b>INSTALACIÓN EN EL SERVIDOR</b> .....	119
<b>1.1    REQUERIMIENTOS MÍNIMOS</b> .....	119
<b>1.2    INSTALACIÓN DEL HARDWARE</b> .....	119
<b>1.3    INSTALACIÓN DEL SOFTWARE</b> .....	119
1.3.1    Gestor de Base de Datos: MySQL .....	120
1.3.2    Entorno de Desarrollo Integrado: Eclipse .....	121
1.3.3    Servidor de Aplicaciones Web: GlassFish v2 .....	124
<b>1.4    CONFIGURACIONES</b> .....	124
2 <b>INSTALACIÓN EN EL CLIENTE</b> .....	125
<b>2.1    REQUERIMIENTOS MÍNIMOS</b> .....	125
<b>2.2    INSTALACIÓN DEL HARDWARE</b> .....	125
<b>2.3    INSTALACIÓN DEL SOFTWARE</b> .....	125
2.3.1    Microsoft Project 2003 .....	125
2.3.2    Componente Addin .....	125
<b>2.4    CONFIGURACIONES</b> .....	126
 <i>ANEXO B. MANUAL DE USUARIO</i> .....	 128
1 <b>INICIO DEL USO DE LA HERRAMIENTA</b> .....	129
2 <b>IDENTIFICACIÓN Y PRUEBA DEL FICHERO WSDL</b> .....	130
3 <b>SELECCIÓN DE PARÁMETROS PARA LA BÚSQUEDA DE PATRONES DE PRODUCTO</b> .....	133
4 <b>SELECCIÓN Y ASIGNACIÓN DE PATRONES DE PRODUCTO</b> .....	135
5 <b>GUARDAR HISTÓRICO DE LA PLANIFICACIÓN</b> .....	137

## ÍNDICE DE FIGURAS

Figura 1. Descripción Gráfica del Proyecto .....	16
Figura 2. Procesos en la Gestión de Proyectos.....	17
Figura 3. WBS estándar .....	20
Figura 4. RBS estándar.....	21
Figura 5. Proceso de Planificación.....	21
Figura 6. Planificación de los Proyectos Software (I).....	22
Figura 7. Planificación de los Proyectos Software (II) .....	22
Figura 8. Ejemplo de la Técnica de Análisis de Riesgos .....	29
Figura 9. Ejemplo 1 - Técnica PERT .....	30
Figura 10. Ejemplo 2 - Técnica PERT .....	31
Figura 11. Plantilla de la Técnica PER .....	31
Figura 12. Ejemplo de la Técnica CPM .....	31
Figura 13. Ejemplo del Diagrama de Gantt.....	32
Figura 14. Ejemplo del Diagrama de Ishikawa .....	33
Figura 15. Ejemplo del Diagrama de Pareto .....	34
Figura 16. Ejemplo del Método Roy - Datos .....	35
Figura 17. Ejemplo del Método Roy.....	35
Figura 18. Diagrama de Casos de Uso de la Aplicación.....	60
Figura 19a. Planificación del Proyecto (I) .....	77
Figura 19b. Planificación del Proyecto (II).....	78
Figura 20. Modelo Conceptual.....	80
Figura 21. Diagrama de Clases .....	83
Figura 22. Diagrama de Secuencia: Identificación .....	84
Figura 23. Diagrama de Secuencia: Probar Descriptor WSDL.....	85
Figura 24. Diagrama de Secuencia: Seleccionar Restricciones .....	85
Figura 25. Diagrama de Secuencia: Realizar Búsqueda de Patrones .....	86
Figura 26. Diagrama de Secuencia: Seleccionar Patrón .....	86
Figura 27. Diagrama de Secuencia: Consultar Información del Patrón.....	87
Figura 28. Diagrama de Secuencia: Seleccionar Tareas Destino.....	87
Figura 29. Diagrama de Secuencia: Crear Columna Destino.....	88
Figura 30. Diagrama de Secuencia: Guardar proyecto en BBDD.....	88
Figura 31. Diseño: Esquema de Comunicación .....	89
Figura 32. Comunicación interna del Web Service.....	92
Figura 33. Leyenda para el Diagrama de Flujo .....	93
Figura 34. Diagrama de Flujo del funcionamiento de la aplicación .....	94
Figura 35. Otra representación del funcionamiento de la aplicación .....	95
Figura 36. Modelo de Datos: Tablas para la búsqueda de Patrones.....	96
Figura 37. Modelo de Datos: Tablas para guardar históricos de proyecto.....	97
Figura 38. Modelo de Datos: Base de datos completa .....	98
Figura 39. Arrancar Servidor GlassFish v2.....	121
Figura 40. Crear Web Service .....	122
Figura 41. Configurar Web Service .....	123
Figura 42. Inicio de Microsoft Project .....	129

Figura 43. Acceso a la herramienta (I) .....	129
Figura 44. Acceso a la herramienta (II).....	130
Figura 45. Conexión y Prueba WSDL .....	130
Figura 46. Asistente para la creación de la Columna Destino (I) .....	131
Figura 47. Asistente para la creación de la Columna Destino (y II) .....	132
Figura 48. Calendario con columna URLWiki recién creada .....	132
Figura 49. Selección del Contexto y las Restricciones .....	133
Figura 50. Selección de los Parámetros de Búsqueda.....	134
Figura 51. Mensaje de estado (I).....	134
Figura 52. Selección de Patrones .....	135
Figura 53. Selección de las Tareas Destino de la actualización.....	136
Figura 54. Mensaje de confirmación (I).....	137
Figura 55. Mensaje de confirmación (II) .....	137
Figura 56. Mensaje de estado (II).....	138
Figura 57. Calendario con columna URLWiki rellena.....	138



# 1 INTRODUCCIÓN

## 1.1 DESCRIPCIÓN GENERAL DEL PROYECTO

La principal razón para realizar el presente Proyecto Fin de Carrera (*PFC*) ha surgido de la motivación por desarrollar una herramienta o aplicación de ayuda que aporte conocimiento en la realización de planificaciones de proyectos de desarrollo software, aprovechando la experiencia de proyectos previos y fomentando la utilización de las Buenas Prácticas de la ingeniería del software en los mismos.

Actualmente, existe un alto porcentaje de fracasos en el desarrollo de proyectos software; algunos de los motivos más destacados son: [Sanchez-ISW2, 2007]

- Objetivos poco claros
- Pobre gestión de proyectos software
- Se trabaja con una Tecnología nueva
- Escaso uso de las Buenas Prácticas de la ingeniería del software
- Recursos (humanos/materiales) insuficientes o mal asignados
- Costes irreales

De esta manera, la herramienta desarrollada tiene como finalidad aumentar dicho porcentaje de éxito, reduciendo el impacto negativo de alguno de los elementos listados previamente, por medio del uso de conocimiento obtenido de proyectos ya desarrollados, así como la aplicación de las Buenas Prácticas a través de un conocimiento almacenado en un artefacto denominado “Patrones de Producto”, cuya definición se puede encontrar en la sección 2.2.1.

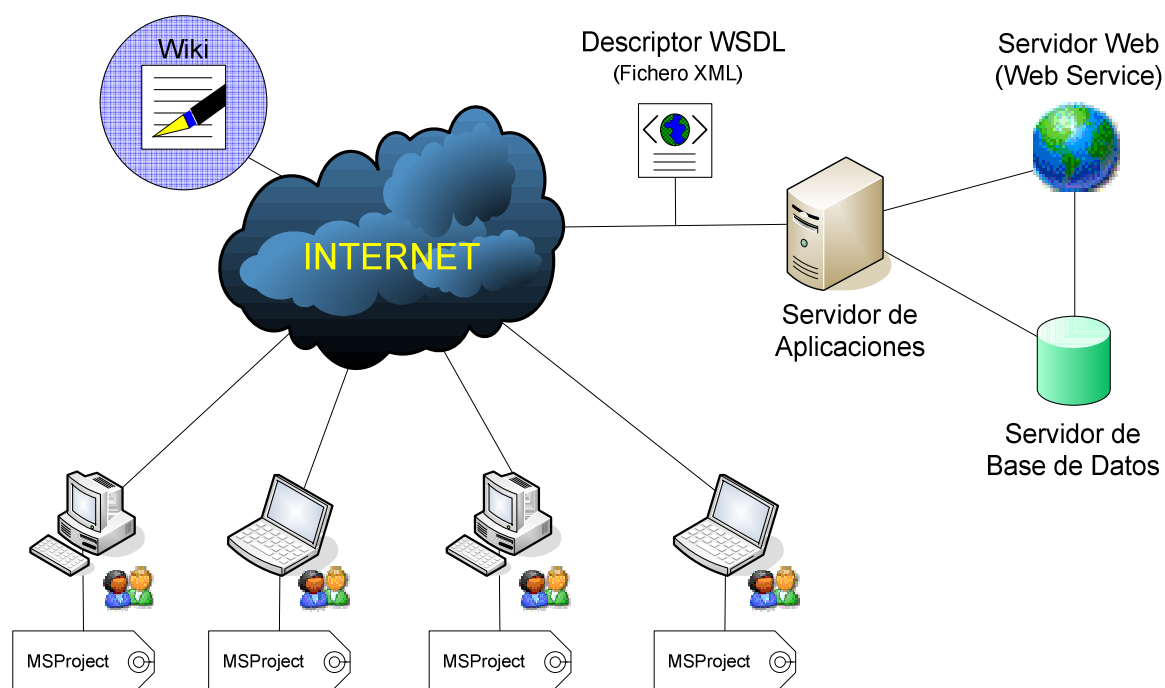
El uso de esta herramienta va dirigido a los Jefes de Proyecto o Analistas, encargados de la planificación de un proyecto software, quienes dispondrán de la capacidad y permisos suficientes para realizar la planificación de proyectos basados en tareas y sus relaciones, la gestión de recursos asociados a las tareas, la evaluación de costes y el seguimiento de las estimaciones plasmadas en los cronogramas, para su posterior validación y/o modificación. A lo largo del presente documento, a dichos usuarios los llamaremos *Jefes de Proyecto* o *JP*.

Para alcanzar el objetivo indicado inicialmente, la presente documentación describe las fases de análisis, diseño e implementación que son definidas para la realización de la aplicación, del modo más claro, legible y objetivo posible, con la intención de que el tribunal, o cualquier otro lector interesado en el tema, ajeno o no al mismo, sea capaz de comprenderlo sin problema alguno.

Avanzando a un nivel más detallado, y menos conceptual que el visto hasta el momento con respecto a la manera sobre cómo se ha llevado a cabo el proyecto actual, se deben comentar brevemente los elementos principales existentes en la herramienta:

- Por un lado, disponemos de un componente software, al que llamaremos *Addin* en el presente documento, el cual se trata de un módulo externo con funcionalidad propia que se añade a un software ya existente, pudiendo hacer uso, o no, de las funcionalidades públicas de este último. En este PFC, el Addin ha sido construido para completarse a MS-Project, sirviendo como interfaz por el cual el Jefe de Proyecto puede interactuar, de manera gráfica, con el sistema y a través del cual, puede realizar peticiones para obtener la información que necesita, como se puede apreciar en el punto 4.3 del documento.
- Por otro lado, disponemos de un Servicio Web (*Web Service*, nombre que utilizaremos de manera habitual durante la presente documentación), el cual tiene como función dar respuesta a las peticiones realizadas por el Jefe de Proyecto a través de la interfaz del componente software o Addin de MS-Project.

La figura 1 muestra la distribución de los elementos necesarios que intervienen dentro de este proyecto:

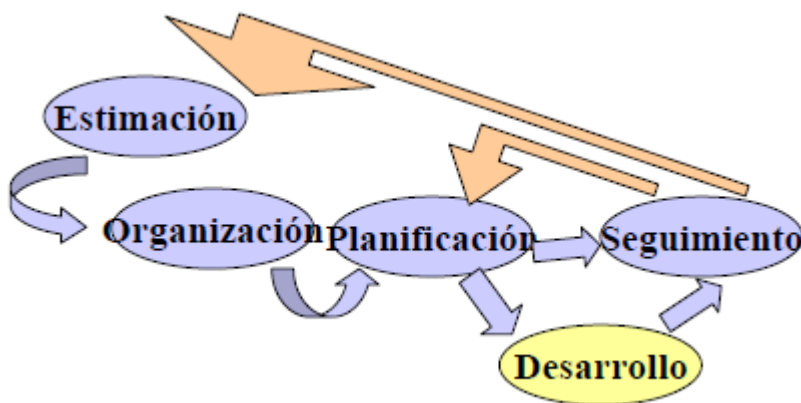


**Figura 1. Descripción Gráfica del Proyecto**

Como se puede observar en la figura 1, los equipos clientes disponen de la aplicación MS-Project instalada, junto el componente Addin. Este componente, a petición del usuario y por medio del interfaz del cliente, realiza las invocaciones de los servicios publicados por el Web Service (por medio del descriptor de servicios WSDL, accedido desde internet). Los servicios reciben los correspondientes parámetros de entrada, realizan las operaciones para las que fueron diseñadas, acceden a las bases de datos con el propósito de consultar o almacenar información, según sea la necesidad, y, tras completar su ejecución, devuelven el resultado.

Además, se puede consultar toda la información referente a los Patrones de Producto aplicables a las planificaciones de los Jefes de Proyecto accediendo, por medio de Internet, a una Wiki pública (<http://productpatterns.sel.inf.uc3m.es>), en la cual se muestran en detalle las características de cada uno de estos Patrones (restricciones, contextos, soluciones, estimaciones de tiempo, tareas y sub-tareas, etc.).





Resumiendo, la aplicación desarrollada para el presente proyecto se halla dentro del Proceso de Planificación del desarrollo software y, tiene el objetivo de ayudar a los Jefes de Proyecto en la aplicación de las Buenas Prácticas, aportando conocimiento a sus calendarios por medio de Patrones de Producto.

### **1.2.2 Procesos/Actividades de la Gestión de Proyectos**

En este apartado se expone detalladamente cada uno de los procesos pertenecientes a la Gestión de Proyectos. De esta manera, se pretende ofrecer una visión más completa sobre los pasos que se deben de seguir para desarrollar un proyecto software correctamente. Estos procesos, y sus características más relevantes, son:

#### **I. Proceso de Estimación: [Sanchez-ISW2, 2007]**

Es la primera etapa en la Gestión de Proyectos y se define como el proceso que proporciona un valor a un conjunto de variables para la realización de un trabajo dentro de un rango aceptable de tolerancia.

Además, es la etapa en la que se realiza la predicción de personal, del esfuerzo, de los costes y del tiempo que se requerirá para realizar todas las actividades y construir todos los productos asociados a un proyecto. Se trata de un proceso continuo que va cambiando a medida que se va conociendo el proyecto.

Es difícil realizar una buena estimación software por diversas razones:

- No existe un modelo universal de estimación o fórmulas universalmente aplicables a todos los proyectos.
- Existen muchas personas implicadas a distintos niveles en los proyectos organizativos, los cuales precisan distintas vistas de la estimación.
- La utilidad de la estimación depende de la etapa del desarrollo.
- Es difícil obtener estimaciones claras y precisas al comienzo de un proyecto.

- La tecnología usada influye directamente en la estimación y hay poco tiempo para adaptarse a los cambios.
- Tendencia a subestimar.
- Malas interpretaciones de las unidades de medida: el software realizado por 25 personas en dos años se puede hacer por 50 personas en 1 año.

A la hora producir información de gestión significativa y a tiempo se hace uso de las llamadas “*Métricas*”, las cuales son la aplicación continua de técnicas basadas en las medidas de los procesos de desarrollo software y sus productos. Existen disponibles varios tipos de Métricas:

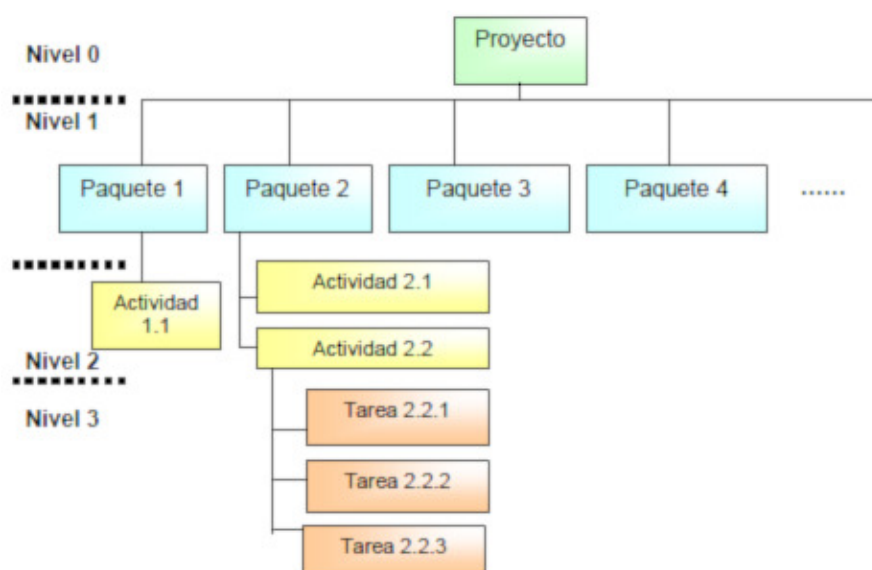
- Métricas de Producto: sirven para estimar el tamaño de las aplicaciones software.
  - Puntos de Función (Albretch, Mark II).
  - Líneas de código.
  - Bang’s (de DeMarco).
- Métricas de Proceso: sirven para estimar el tiempo de desarrollo y el esfuerzo. Implican el uso de alguna técnica:
  - Basadas en opinión de expertos: basado en la experiencia profesional de los participantes.
  - Analogías: comparación directa con proyectos pasados.
  - Descomposición: descomponer el producto y la estimación global es la suma de las estimaciones de los componentes.
  - Ecuaciones de estimación (Modelos estadísticos, basados en teorías, modelos compuestos -COCOMO, SOFTCOST, SPQR, COPMO, ESTIMACS-).

## II. Proceso de Organización: [Sanchez-ISW2, 2007]

El proceso de organización de un proyecto software se compone de varios pasos:

- Identificar los hitos y entregables del proyecto.
- Descomponer el proyecto en grupos de actividades elementales hasta constituir el árbol de descomposición de actividades.
- Construir la estructura de los equipos de desarrollo.
- Construir los organigramas de los equipos de realización.
- Distribuir el presupuesto entre las actividades y fases. Asignación de costes para “Actividades” y para “Partidas”.
- Elegir modelo de ciclo de vida o modelo de proceso, metodología de desarrollo, etc.

Para realizar la estructuración de Tareas se utiliza una estructura exhaustiva, jerárquica y descendente llamada WBS (Work Breakdown Structure) formada por los entregables a realizar en un proyecto, compuesto por varios niveles, según de detalle final que se pretenda mostrar. En la figura 3 se muestra dicha estructura WBS estándar:

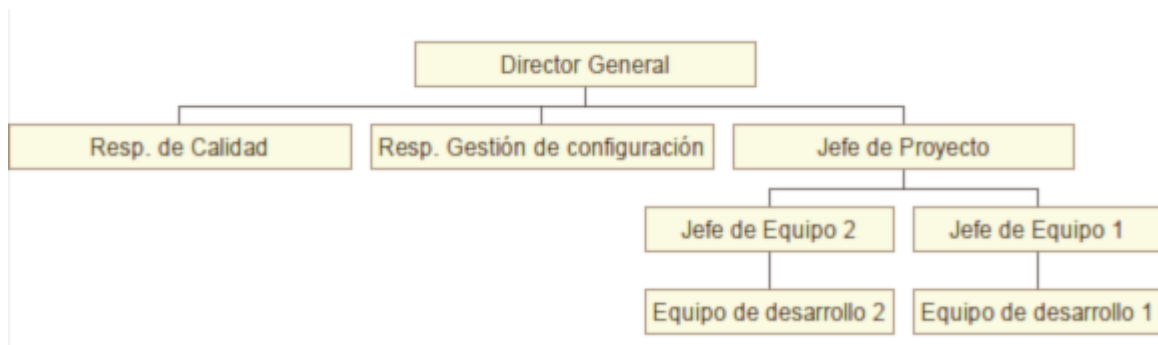


**Figura 3. WBS estándar**

Por otro lado, para representar los recursos humanos y materiales del proyecto se presenta una estructura de árbol jerárquica, llamada RBS (Resource Breakdown Structure), cuyos principales objetivos son los siguientes:

- Mostrar gráficamente la organización humana del proyecto.
- Maximizar el uso de los conocimientos y experiencia del personal disponible.
- Reflejar la estructura de recursos materiales necesarios para la realización del proyecto, así como sus costes asociados.

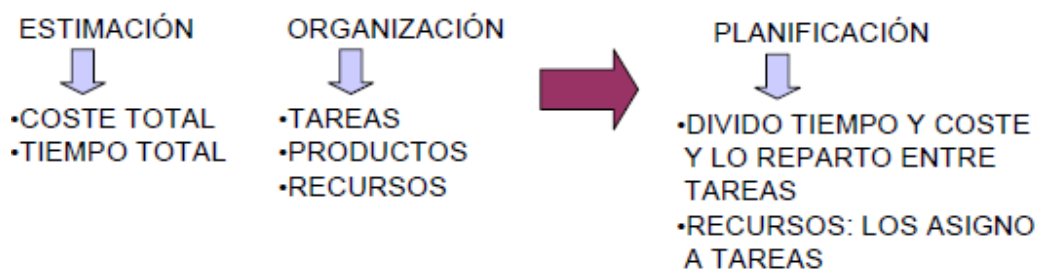
En la figura 4 se muestra una estructura RBS estándar:



**Figura 4. RBS estándar**

### III. Proceso de Planificación: [Sanchez-ISW2, 2007]

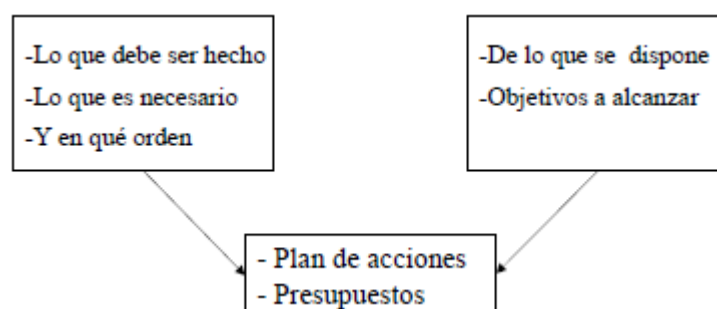
El Proceso de Planificación toma su entrada de los Procesos de Estimación y de Organización, y se define como el proceso de selección de una estrategia para la obtención de los productos software finales dados. La figura 5 muestra esquema del proceso:



**Figura 5. Proceso de Planificación**

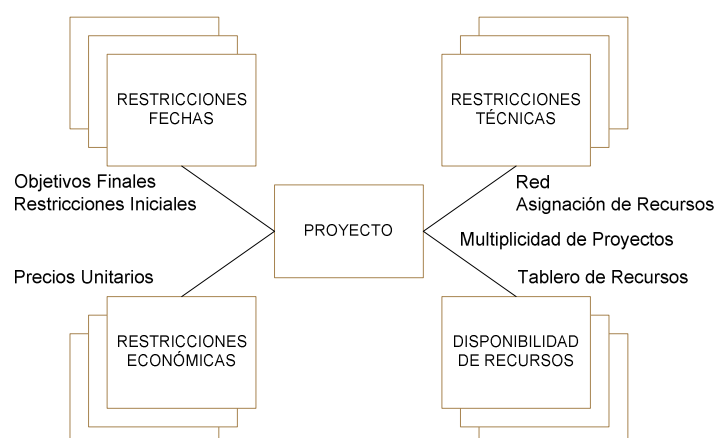
Es importante realizar una buena planificación de los proyectos para evitar el fracaso de los mismos. Por esta razón, es importante tener en cuenta los objetivos dentro esta etapa, los cuales son:

- Identificar de una forma precisa lo que se debe hacer.
- Identificar la secuencia de realización.
- Establecer la coherencia con los recursos disponibles.
- Transformar el modelo teórico del proyecto en un plan de acción aplicable.



**Figura 6. Planificación de los Proyectos Software (I)**

Un proyecto es un conjunto de actividades que deben satisfacer una serie de restricciones existentes, además de alcanzar un objetivo pre-establecido, por lo que se deben tener en cuenta múltiples elementos a la hora de realizar una correcta planificación. En la figura 7 se muestra un ejemplo de planificación de proyectos software:



**Figura 7. Planificación de los Proyectos Software (II)**

En la sección 1.5 se listan varias de las técnicas aplicables durante el Proceso de Planificación (Análisis de redes -Network Analysis-, Método del Camino Crítico -CPM-, PERT -Programme Evaluation Review Technique-, Gantt, etc.), y en la sección 1.6 se listan varias de las herramientas existentes que hacen uso de algunas de esas técnicas (MSProject, Office manager, OpenProj, etc.).

#### IV. Proceso de Seguimiento: [Sanchez-ISW2, 2007]

La etapa de Seguimiento de un proyecto software tiene el objetivo de obtener información acerca del progreso, estado y trayectoria de dicho proyecto. De este modo, facilita una visión adecuada del progreso real, de forma que los Jefes de Proyecto puedan tomar medidas eficaces cuando el desarrollo del proyecto software se desvíe notablemente de lo planificado, para ello es necesario establecer marcas de seguimiento (hitos). El Proceso de Seguimiento se compone de las siguientes fases:

- Obtención de información.
- Análisis.
- Comparación.
- Acción correctora (si procede).

Los Objetivos que se deben cumplir para alcanzar el éxito son:

- Identificar diferencias entre lo planificado y lo realizado.
- Evaluar el avance del proyecto.
- Adaptar el plan de acción a las diferencias encontradas.
- Prever desviaciones importantes para buscar remedios.
- Contribuir a la creación de históricos.
- Contabilizar costes de cada actividad.

Su aplicación se puede llevar a cabo con diferente periodicidad, dependiendo de las necesidades del proyecto. Esta periodicidad puede ser:

- Inmediatamente después de hacer la planificación detallada.
- Periódicamente, donde se recogen medidas de avance y se analizan las diferencias.
- En periodos más grandes se revisan desviaciones y se decide cómo afectan al Plan de Proyecto definido.
- Al alcanzar ciertos Hitos importantes, donde se revisa el progreso y se toman medidas correctivas, si fueran necesarias.

Existen varios tipos de Seguimiento de proyectos que se deben diferenciar, condicionados por el elemento al que se le dé más importancia a la hora de establecer su revisión:

- En función de la duración
  - Seguimiento a corto plazo (cada semana).
  - Seguimiento a largo plazo (cada mes).
- En función de la actuación
  - Estático: mediante anotaciones en fichas de seguimiento.
  - Dinámico: Por medio de la aplicación de la técnica de Gantt de seguimiento.
  - Predictivo: Por medio de la aplicación de técnicas (Diagrama de 45 grados, valor conseguido).

#### V. Proceso de Garantía de la Calidad: [INET, 34], [INET, 35]

Se define como proceso de Garantía de la Calidad al conjunto de acciones planificadas y sistemáticas necesarias para proporcionar la confianza adecuada de que un producto o servicio satisfará los requerimientos dados sobre calidad.

Su misión no se limita a actividades de verificación, sino que además asume un rol de liderazgo en la gestión de la calidad durante el proceso de creación y diseño del producto. La garantía de calidad no debe confundirse con la técnica específica de control de calidad, cuyo objetivo es verificar el producto.



El beneficio principal de un programa de garantía de calidad de software es asegurar a la gerencia del proyecto que los procesos establecidos se han ejecutado adecuadamente. Esta evaluación suele estar realizada por un grupo independiente, especializado en métodos de calidad, con un criterio objetivo y con visión de contexto.

La Garantía de Calidad debe asegurarse de lo siguiente:

- Se usa la metodología de desarrollo apropiada.
- Las actividades de desarrollo han sido debidamente planeadas.
- Se han definido estándares y procedimientos para el proyecto.
- El personal ha sido debidamente entrenado en los procesos de calidad aplicables.
- El desarrollo es documentado adecuadamente para facilitar la mantención y la reutilización.
- La documentación se produce oportunamente y no después que el desarrollo ha sido completado.
- Los cambios introducidos han sido debidamente controlados.
- Las pruebas efectuadas son eficaces para detectar defectos, especialmente en aquellas áreas de mayor riesgo.
- Las actividades se llevan a cabo de acuerdo a los plazos y en los términos planeados.
- Las desviaciones a los estándares se identifican rápidamente.
- El proyecto está en condiciones para ser sometido a auditorías externas, si corresponde.
- La calidad es verificada con respecto a criterios preestablecidos.
- La gerencia es oportunamente informada de problemas y riesgos relativos a la calidad.
- Los problemas de calidad se analizan y las causas se comunican al proyecto para tomar medidas preventivas que eviten su repetición.

Además, la calidad en el software está compuesta por 3 dimensiones como son: el sistema de calidad, la aplicación adecuada del proceso y las actividades que aseguran la calidad del producto.

Con el propósito de realizar la evaluación y mejora de los procesos software existen varios estándares y modelos que pueden ser aplicados, como por ejemplo: ISO 9000 (ISO 9001:2000), (SPICE) ISO/IEC 15504, CMM, Organismos de calidad y/o Certificación en calidad.

### **1.3 EL PROCESO DE PLANIFICACIÓN**

Las aplicaciones de gestión de proyectos están diseñadas para asistir a los Jefes de Proyecto en el desarrollo de la planificación de tareas, la asignación de recursos a las tareas, seguimiento de la planificación diseñada, administración de presupuestos y costes, el análisis de cargas y los balanceos de trabajo a lo largo de las fases de planificación, ejecución y entrega o puesta en marcha del proyecto, con el objetivo de alcanzar los resultados más eficientes en costes, plazos y calidad.

Estas herramientas utilizan técnicas específicas, como por ejemplo, las técnicas de duración mínima PERT (maneja tiempos inciertos de las actividades del proyecto), CPM (maneja tiempos conocidos de las actividades del proyecto) o PDM/AON (técnica de las precedencias), y pueden usarse de manera individual o combinadas entre sí, con el objetivo de beneficiarse de una mayor visión e información durante todo el proceso en la gestión de los proyectos.

La mayoría de las herramientas de gestión de proyectos nos permiten [INET, 01]:

- Hacer visible el progreso de las tareas de un proyecto de manera simple y concisa.
- Planificar sus necesidades en recursos humanos y materiales.
- Ayudar en la construcción del presupuesto del proyecto.
- Seguir los costes de proyecto: consumo de recursos humanos J/h, equipos, etc....
- Comunicar las modificaciones temporales del proyecto.

Todas las empresas, no necesariamente pertenecientes al ámbito informático como lo son, por poner un ejemplo, el de la construcción naval o civil, necesitan ser cada día más y más competitivas, para lo cual es recomendable la realización de una buena planificación y administración de los proyectos de desarrollo y/o de implantación para obtener una eficiencia óptima en cuestión de recursos, plazos y costes y, de este modo, incrementar la productividad de sus operaciones.

Por lo tanto, los objetivos de la planificación y programación de proyectos se pueden resumir en los siguientes: [Recio-GO, 2007]

- Minimizar la duración del proyecto.
- Minimizar el coste del proyecto.
- Minimizar el consumo de recursos.
- Asegurar plazos y calidad.

## **1.4 MOTIVACIÓN**

Una mala gestión en la planificación de los proyectos software desemboca en el fracaso del proyecto, en un exceso de costos y en retrasos en la entrega. Las causas pueden radicar en las omisiones realizadas durante el desarrollo de sistemas, definición imprecisa de objetivos, estimaciones de costos prematuras, deficientes técnicas de estimación, mala gestión de tiempo y falta de liderazgo. Es responsabilidad del Jefe de Proyecto evitar estos errores y llevar a buen término el proyecto, tanto en tiempo como en presupuesto. Entre las funciones básicas del Jefe de Proyecto se incluyen la planificación de las tareas de proyecto, la elección del equipo de proyecto, la organización y la planificación de los esfuerzos del proyecto, la dirección del equipo y el control de la evaluación del proyecto. [INET, 23]

Las herramientas actuales para la gestión de los proyectos son, hoy en día, un aspecto muy importante a tener en cuenta a la hora de obtener los mejores resultados, aportando a los Jefes de Proyecto, mecanismos de control, seguimiento y reacción para evitar problemas indeseados durante el desarrollo de los productos.

Por lo tanto, se deben conocer bien las necesidades existentes para seleccionar una herramienta u otra, así como la técnica o técnicas a emplear, tratando de ser realistas y objetivos en la asignación de recursos y presupuestos con respecto a las tareas y/o sub-tareas que forman los proyectos.

## 1.5 METODOLOGÍAS Y TÉCNICAS PARA EL PROCESO DE PLANIFICACIÓN

Existen varias técnicas que pueden ser usadas durante el proceso de planificación de proyectos y que se ocupan del problema desde diferentes puntos de vista, de manera que, podamos tener una visión más completa y verosímil del entorno, así como de las circunstancias que lo rodean.

Una buena planificación, control y seguimiento de los proyectos, pueden evitar que surjan problemas importantes (sobrecostes, finalización fuera de plazos, falta de recursos, etc....) y permiten alcanzar el éxito del proyecto. Por esta razón, resulta necesario tener en cuenta las características específicas de las diferentes técnicas existentes para hacer uso de aquellas que verdaderamente necesitamos, de modo que se pueda realizar una correcta gestión de los proyectos y potenciando las posibilidades de éxito.

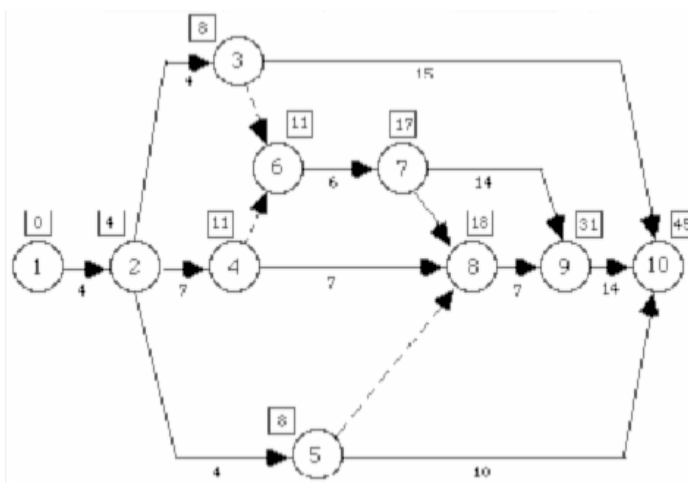
De entre las técnicas más usadas en la actualidad destacamos las más importantes:

- **Análisis de Redes (Network Analysis).** [Sanchez-ISW2, 2007]

Se trata de una técnica mediante la cual se identifican las dependencias entre tareas, cada una de ellas representada por un círculo, con los que se construye una red de tareas y se identifican las tolerancias en tiempo.

El Análisis de Redes es un término colectivo para dos técnicas diferentes a la planificación de proyectos: CPM o “*Critical Path Method*” y “*PERT*”.

Tarea	Duración
1	0
2	4
3	4
4	7
5	4
6	0
7	6
8	7
9	7
10	10



**Figura 8. Ejemplo de la Técnica de Análisis de Riesgos**

- **Método PERT.**

También conocido como “*Gráfico PERT*”, es una técnica que permite *planificar, programar y controlar* los recursos de los que se dispone, con el fin de obtener los resultados deseados [INET, 02] y que se basa, fundamentalmente, en técnicas probabilísticas. [Sanchez-ISW2, 2007]

Consiste en la representación gráfica de una red de tareas que maneja tiempos inciertos que, cuando se colocan en una cadena, permiten alcanzar los objetivos del proyecto, de forma objetiva, sencilla y práctica. Consta de los siguientes elementos: [INET, 03]

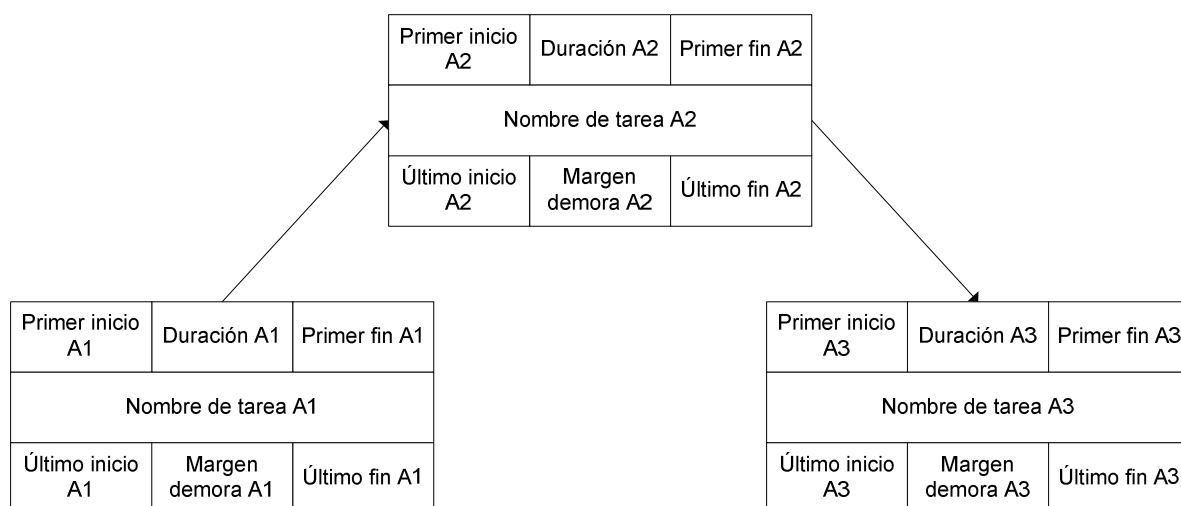
- Tareas
- Etapas
- Tareas ficticias

El uso y las ventajas principales del gráfico PERT se obtienen por su alta capacidad en asistir al Jefe de Proyecto en la planificación y el control de los mismos [INET, 04] y es adecuado para procesos repetibles y completamente dominados (producción, construcción, manufactura, etc...). [Sanchez-ISW2, 2007]

Por el contrario, también tiene una serie de desventajas y de problemas que surgen cuando están involucrados nuevas tecnologías y grandes grupos humanos en la planificación de proyectos, de donde pueden aparecer complicaciones con duraciones no realistas, tareas y/o enlaces no previstos. Después de un cierto tiempo la planificación puede ser considerada inútil, ya que la red es muy estática y no permite realizar ajustes y modificaciones de forma sencilla. [Sanchez-ISW2, 2007]

Esta técnica tiene en cuenta la *Ruta Crítica*, *Camino Crítico* o secuencia de eventos más larga que determina cuanto puede tardar en completarse un proyecto, como máximo. La Ruta Crítica es una pieza clave de control de los proyectos (las actividades de la ruta crítica son tareas que de tardarse más de lo programado en finalizar, retrasarían la planificación total del proyecto). [Sanchez-ISW2, 2007]

Para el resto de las tareas que no están en el camino crítico existe un grado de tolerancia (comienzo más tardío, finalización más tardía, comienzo más temprano, etc....). [Sanchez-ISW2, 2007]



**Figura 9. Ejemplo 1 – Técnica PERT**

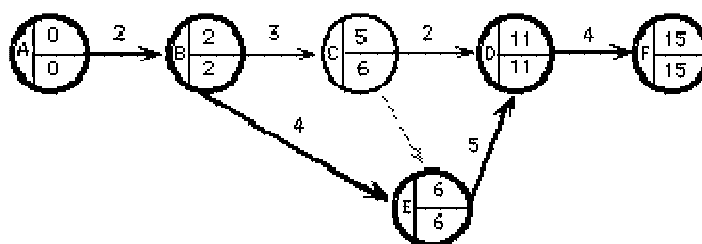


Figura 10. Ejemplo 2 - Técnica PERT

Primer inicio	Duración	Primer fin
Nombre de tarea		
Último inicio	Margen demora	Último fin

Figura 11. Plantilla de la Técnica PER

- **Método CPM** (Critical Path Method).

Es idéntico al método PERT en concepto y metodología aunque la principal diferencia existente entre ellos es, simplemente, la técnica por medio de la cual se realizan los estimados de tiempo para las actividades [INET, 04]: CPM asume que los tiempos de cada actividad son conocidos con certeza y por tanto se tiene un solo tiempo para cada actividad [INET, 05].

Al igual que PERT, es una técnica que tiene en cuenta la *Ruta Crítica*.

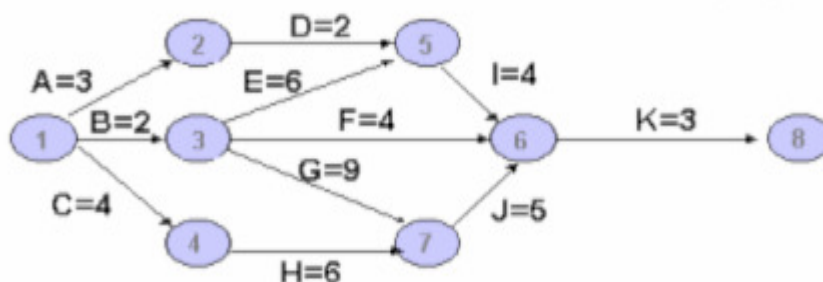


Figura 12. Ejemplo de la Técnica CPM

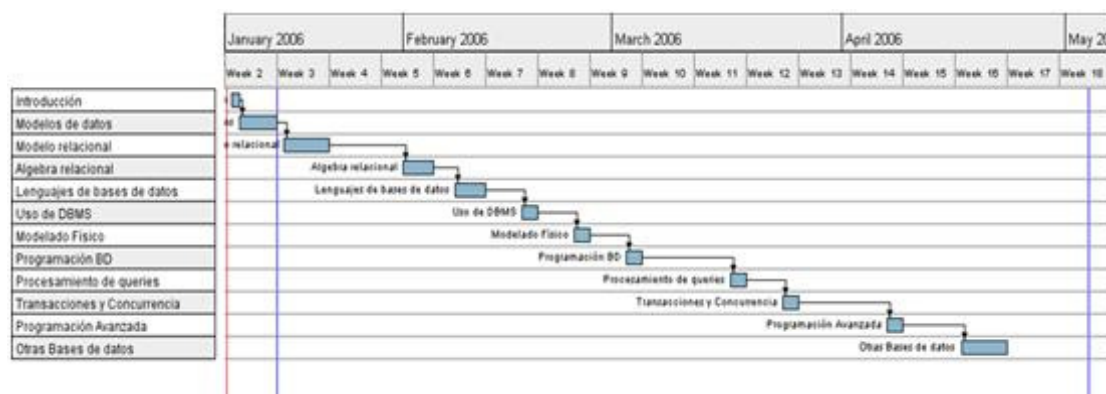
- **Diagramas Gantt.**

Es una sencilla técnica de gráficos de tiempos que resultan bastante eficaces para la planificación y la evaluación del avance de los proyectos. Se trata de un sencillo gráfico de barras (cada una ellas simboliza una tarea del proyecto), encadenadas entre si y dispuestas en un eje horizontal que representa el tiempo.

Una ventaja importante de los gráficos Gantt es que ilustran claramente el solapamiento entre tareas planificadas [INET, 04] y es una gran ayuda memorística para saber la ubicación temporal del proyecto [Sanchez-ISW2, 2007].

Sin embargo, se trata de la representación estática de una situación dinámica. En concreto, no se muestran las interdependencias entre tareas (cada una representada por una barra), y las cuales tienen que estar presente en la mente del Jefes de Proyecto al dibujar o utilizar este tipo de gráficos. [Sanchez-ISW2, 2007]

Los diagramas Gantt se utilizan a menudo como instrumento de comparación para medir el progreso de un proyecto, y los Jefes de Proyecto suelen pasarlo como información a los jefes de equipo. [Sanchez-ISW2, 2007]



**Figura 13. Ejemplo del Diagrama de Gantt**



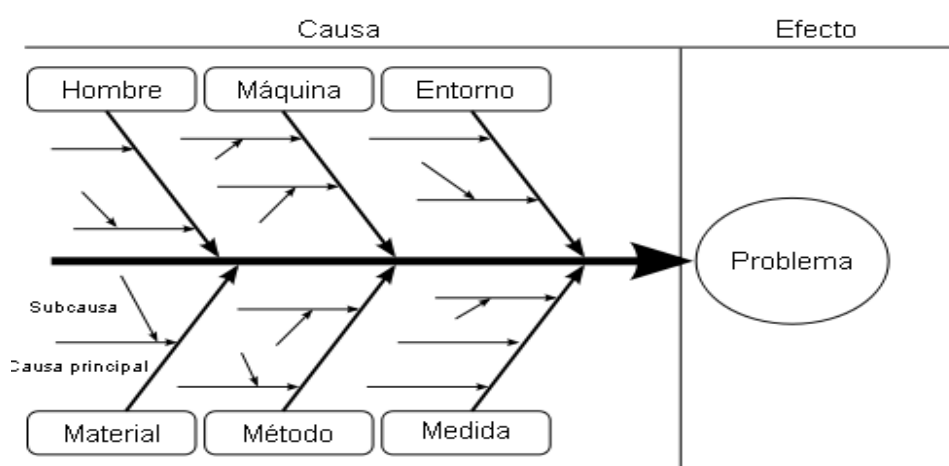
Además de las comentadas, existen otro tipo de técnicas que pueden ser útiles y que nos pueden ofrecer una visión complementaria durante la gestión de la planificación. Entre las múltiples opciones destacamos:

- **Diagramas de Ishikawa.**

También conocida como *Diagrama de Causa-Efecto*, con origen en el ámbito de la industria y posteriormente en el de los servicios, que sirve para facilitar el análisis de problemas y sus soluciones en esferas como es la calidad de los procesos, los productos y servicios.

Consiste en un diagrama que, por su estructura ha venido a llamarse también *Diagrama de Espina de Pescado*, consistente en una representación gráfica sencilla en la que puede verse de manera relacional una especie de espina central, que es una línea en el plano horizontal, representando el problema a analizar. [INET, 06]

Este tipo de herramienta permite un análisis participativo mediante grupos de mejora o grupos de análisis, que mediante técnicas como por ejemplo la lluvia de ideas, sesiones de creatividad, y otras, facilita un resultado óptimo en el entendimiento de las causas que originan un problema, con lo que puede ser posible la solución del mismo. [INET, 07]

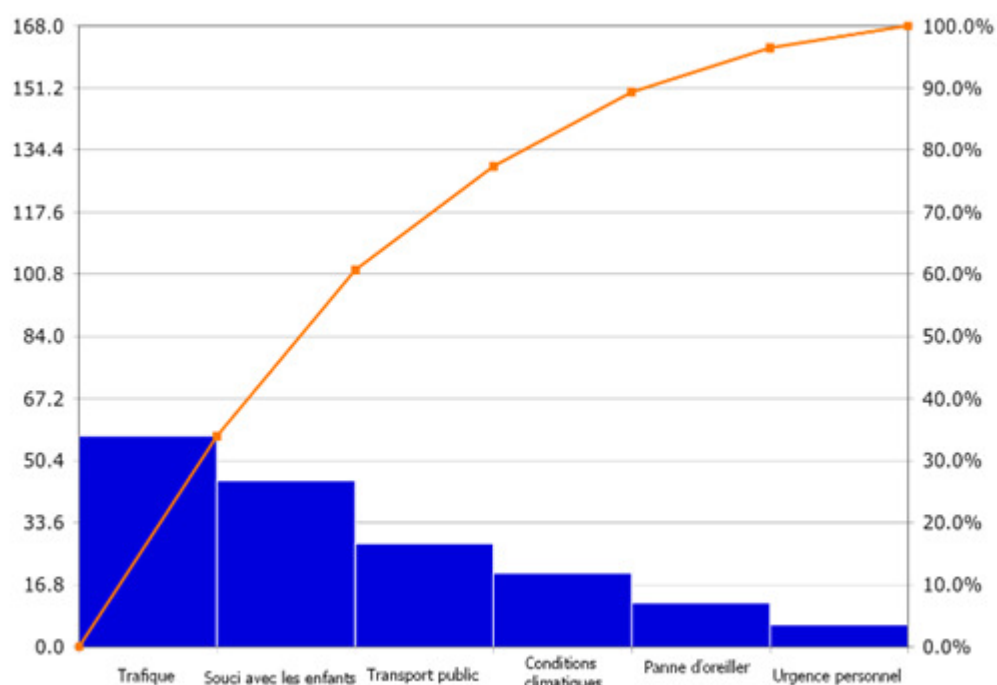


**Figura 14. Ejemplo del Diagrama de Ishikawa**

- **Diagrama de Pareto.** [INET, 08]

También llamada “*Curva 80-20*” o “*Distribución A-B-C*”. Se trata de una gráfica para organizar datos, de forma que estos queden en orden descendente, de izquierda a derecha, y separados por barras. Permite, pues, asignar un orden de prioridades.

El diagrama permite mostrar gráficamente el principio de Pareto (pocos vitales, muchos triviales), es decir, que hay muchos problemas sin importancia (a la derecha de la gráfica) frente a unos pocos graves (a la izquierda de la gráfica).



**Figura 15. Ejemplo del Diagrama de Pareto**

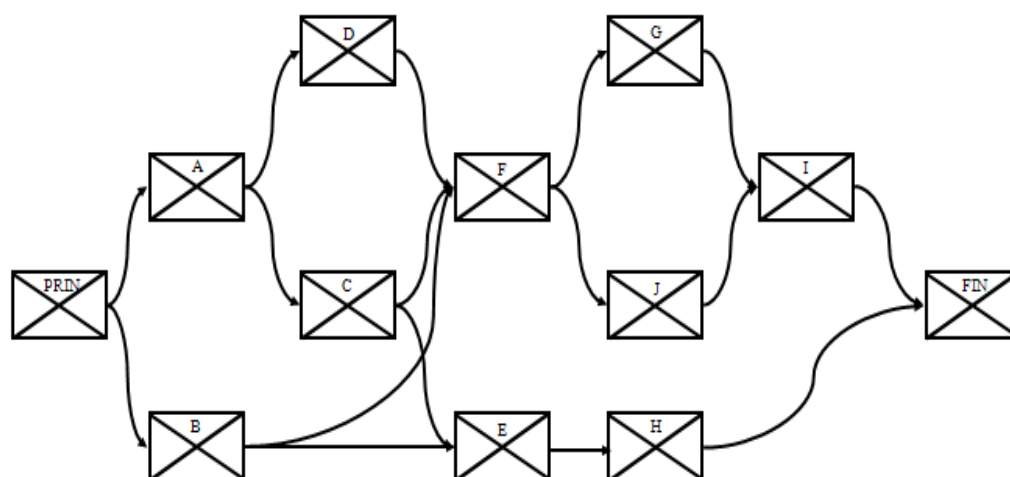
- **Método Roy.**

Se le conoce también como el “*Método de los Potenciales*” o “*MPM*” y, a semejanza del Pert, ofrece una serie de planes alternativos y suele complementarse con algún sistema gráfico a efectos de representación. Por el contrario, se diferencia del PERT/CPM básicamente en dos aspectos: en su construcción y en el tipo de relaciones que se pueden manejar entre actividades. [INET, 09]

Se trata de una técnica sencilla, idónea para proyectos complejos, y que proporciona varios planes de ejecución aunque solo admite relaciones del tipo final/comienzo y comienzo/comienzo, con demora. [INET, 10] Es conveniente utilizar un método de representación gráfica como complemento al método Roy. [INET, 09]

Actividad	Actividad Precedente	Actividad Siguierte
A	-	C,D
B	-	E,F
C	A	E,F
D	A	F
E	B,C	H
F	B,C,D	G,J
G	F	I
H	E	-
I	G,J	-
J	F	I

**Figura 16. Ejemplo del Método Roy - Datos**



**Figura 17. Ejemplo del Método Roy**

## 1.6 HERRAMIENTAS EXISTENTES PARA EL PROCESO DE PLANIFICACIÓN

En el mercado actual conviven diferentes herramientas software para la gestión de proyectos, tanto gratuitas (Open Source o código abierto) como de pago (o por licencias), las cuales sirven de soporte a los Jefes de Proyecto en sus planificaciones. Cada una de estas herramientas dispone de sus propias características, pero todas ellas alineadas sobre una misma filosofía: gestionar el mayor número de aspectos relevantes (tareas, actividades, relaciones entre tareas, recursos, costes, etc....) a la hora de realizar la planificación de proyectos para obtener el éxito final del mismo.

Además, estas herramientas cumplen una serie de requisitos básicos: son fáciles de usar, su manejo es sencillo de recordar después de un tiempo sin ser utilizadas, son claras mostrando resultados o información relativa al proyecto y su gestión (bajo demanda o no del Jefe de Proyecto), son intuitivas, concisas, precisas y eficientes.

Algunas de las herramientas de pago (o por licencias) son:

- **Microsoft Project o MSP**

Herramienta software muy útil para la gestión de proyectos que aplica procedimientos descritos en el PMBOK (Management Body of Knowledge) del PMI (Project Management Institute) [INET, 11], propiedad de Microsoft, con la finalidad de asistir a los Jefes de Proyecto en el desarrollo de planes, asignación de recursos a tareas, seguimiento al progreso, administración del presupuesto y análisis de las cargas de trabajo. [INET, 12]

Actualmente es el referente por excelencia de este tipo de aplicaciones y tiene la gran ventaja de que se integra con el resto de herramientas de MS-Office, lo que hace que su potencial sea aun mayor al poder combinarse con el resto de estas herramientas.

- **Microsoft Visio**

Software de dibujo vectorial, propiedad de Microsoft, que permite realizar, entre otras cosas, diagramas de oficinas, diagramas de bases de datos, diagramas de flujo de programas y representación UML [INET, 13].

Se trata de una herramienta sencilla e intuitiva que, al igual que ocurre con MSP, se integra con el resto de herramientas de MS-Office.

- **Office manager**

Herramienta software de gestión de proyectos bajo una concepción ERP-CRM-SMC [INET, 14].

Algunas de las herramientas gratuitas (Open Source o código abierto) son:

- **OpenProj** [INET, 15]

Herramienta software libre para la gestión de proyectos que permite crear, entre otros, diagramas de Gantt, PERT, WBS, RBS, costes, etc....

- **KMKey** [INET, 16]

Herramienta software para la gestión de proyectos en entorno web.

Integra tres niveles:

- KMKey Project: para la planificación, gestión y control de sus proyectos.
- KMKey Quality: integra el Sistema de Gestión de la Calidad de la organización.
- KMKey HelpDesk: donde apoyar el servicio de atención al cliente y mantenimiento.

- **Kplato** [INET, 15]

Es la herramienta de gestión de proyectos de la suite ofimática Koffice del proyecto KDE. Como componente de Koffice se puede usar integrado con otros componentes de Koffice.

Usa las librerías QT. Características:

- Usa diagramas de Gantt para mostrar la temporalización de las tareas.
- Permite usar dependencias entre tareas.
- Usa estructuras de desglose de trabajo en forma de árbol.

- **GanttProject**

Herramienta para crear una completa planificación de un proyecto de forma muy visual, similar a MS-Project. [INET, 17], aunque mucho más ligera, que permite programar y organizar las tareas y asignación de personas y recursos sobre una representación Gantt. [INET, 18]

- **Planner** [INET, 19]

Herramienta de gestión de proyectos (planificación, temporización y seguimiento), que permite definir tareas y sub-tareas, recursos y grupos de recursos, dependencias, mostrar el camino crítico, diagramas Gantt, etc....

- **DotProject**

Herramienta escrita en PHP, tipo “Project Manager”, pero para ser utilizado en web [INET, 19] que nos ofrece un marco completo para la planificación, gestión y seguimiento de múltiples proyectos para clientes diferentes, quienes pueden disponer también de acceso para monitorizar la evolución del desarrollo. [INET, 18]

- **TaskJuggler** [INET, 20]

Alternativa al Planner dentro del entorno KDE.

- **Dia** [INET, 21]

Se trata de un editor de diagramas creado como respuesta Open Source a Visio, la herramienta de Microsoft, además de ser totalmente compatible con ella.

Es parte del proyecto GNOME, el popular escritorio de Linux, y es una aplicación GPL.

- **Open Workbench** [INET, 22]

Es otra herramienta de software libre, dirigida al sistema operativo Windows, y que ofrece una robusta funcionalidad para la gestión de proyectos en la empresa, permitiendo el diseño de programas, calendarización, administración de tiempos, entre otros.

- **TeamWork** [INET, 18]

Herramienta que usa apariencia de entorno web para registrar y gestionar los tiempos de diferentes equipos de trabajo en sus respectivos proyectos. Lleva a cabo una gestión completa de los informes de tiempos y costes, además de combinar la gestión de documentos, de equipos y de proyectos.

- **AgileTrack** [INET, 18]

Herramienta diseñada para realizar la planificación y seguimiento de proyectos por medio de un sencillo interfaz, usado en el desarrollo de software en equipos reducidos con metodologías ágiles, especialmente eXtreme Programming.

- **PPTS** [INET, 18]

Project Planning and Tracking System (PPTS) es una herramienta de gestión ágil de proyectos para equipos que trabajan con Scrum y/o Extreme Programming. Es un sistema web, accesible con un navegador que puede instalarse sobre servidor Linux o Windows (con PHP y MySQL) y de uso libre, con licencia GNU (GPL).

- **XPWeb** [INET, 18]

Plataforma web para gestión de proyectos con Extreme Programming.

- **Trac** [INET, 18]

Plataforma web para comunicación, gestión y seguimiento de proyectos, que integra un wiki, interfaz de subversión para la gestión de versiones, seguimiento de proyecto y sistema de tickets para gestionar y registrar tareas, bugs, etc.

- **TUTOS** [INET, 18]

Herramienta web de código abierto y uso gratuito para la gestión de pequeños grupos de trabajo o departamentos. Incluye calendario, gestión de equipos, directorio de personas, gestión de incidencias, registros de tiempo, listas de seguimiento...

- **SoloDox** [INET, 18]

Servicio de software que permite editar y compartir planificaciones Gantt.

- **Mindquarry** [INET, 18]

Herramienta basada en web para la gestión de grupos de trabajo, entornos colaborativos, proyectos... que pretende ser la alternativa open source de soluciones propietarias como Basecamp o Sharepoint.



- **X-Man** [INET, 18]

X-Man (Extreme Manager), es una herramienta fácil para gestión y seguimiento de proyectos ágiles. Se trata de un fichero de 4 Mb que no necesita instalación.

- **ToDoList** [INET, 18]

Herramienta gratuita muy simple y efectiva para la gestión de proyectos en entornos ágiles. Escasamente ocupa 1 Mb y se puede configurar durante el proceso de instalación para guardar la información de configuración (no modifica el registro de Windows).

- **Clocking IT** [INET, 18]

Es un gestor de proyectos y tareas, con control de tiempos, generador de informes, repositorio de ficheros, agenda, chat y notificaciones.

- **IceScrum** [INET, 18]

Con el mismo interfaz para todos los roles, ofrece las opciones de operación, consulta, estimación de historias de usuario... activas o no, según el usuario sea propietario del producto, gestor, equipo o interesado.

También incluye listas de historias de usuario (backlog), de asuntos, de problemas y de pruebas, así como un chat en línea.

- **Gantt PV** [INET, 18]

Herramienta gratuita, de apariencia sencilla y sin grandes complicaciones, para planificación de proyectos, descomposición, representación y seguimiento de tareas sobre diagrama de Gantt.

- **Otras Herramientas software:** [INET, 18]

Project Dune, Activity Manager, PrjPlanner, Project2Manage, Collabtive, RedMine, Sprintometer, Pivotal Tracker, etc....

## 1.7 OBJETIVOS

El objetivo principal de este PFC es realizar el análisis, diseño, implementación y documentación de una herramienta que aporte conocimiento a los Jefes de Proyecto durante la fase de planificación de proyectos software, reutilizando conocimiento de proyectos software previos y aplicando las Buenas Prácticas para el proceso. Este conocimiento reutilizable ha sido previamente analizado, estructurado y almacenado dentro de los Patrones de Producto.

Así pues, las fases y tareas llevadas a cabo durante este proyecto son las que muestran a continuación:

- Planificación y Especificación de Requisitos
  - Especificación de Requisitos Software. (Std. IEE-830)  
(Sección 2.3)
  - Diagramas de Casos de Uso.  
(Sección 2.4.2)
  - Casos de Uso de Alto Nivel.  
(Sección 2.4.3)
  - Definición de los Casos de Uso en Formato Expandido.  
(Sección 2.4.4)
  - Planificación de las Actividades realizadas y las horas imputadas a cada una de ellas.  
(Sección 2.5)

- Análisis
  - Modelo Conceptual.  
(Sección 3.1)
  
- Diseño
  - Diagrama de Clases de Diseño.  
(Sección 4.1)
  - Diagramas de Secuencia.  
(Sección 4.2)
  - Definición de la Interfaz de Usuario.  
(Secciones 4.3, 4.4, 4.5)
  - Diagrama de Flujo.  
(Sección 4.6)
  - Esquema de la Base de Datos.  
(Sección 4.7)
  
- Implementación y Pruebas
  - Desarrollo de todos los Casos de Uso.  
(Sección 5 – Ejemplo de implementación de un Caso de Uso)

## **2 PLANIFICACIÓN Y ESPECIFICACIÓN DE REQUISITOS SOFTWARE**

### **2.1 INTRODUCCIÓN**

Como se menciona en [Pressman, 1998], la especificación de los requisitos del software implica la culminación de la tarea del análisis de la aplicación. Dicha especificación se logra estableciendo una completa descripción de las clases que colaboran, su función y el comportamiento de la aplicación, de forma que se tenga una idea clara y concisa de lo que hay que realizar, quedando satisfechas las necesidades de quien demanda la construcción del sistema.

Este apartado cumple tres objetivos importantes:

- Describir lo que requiere el usuario.
- Establecer una base para la creación de un diseño de software.
- Definir un conjunto de requisitos que se puedan validar una vez que se ha construido el software.

De esta manera, se ha tratado de establecer las bases para un buen diseño de la aplicación, documentando la funcionalidad del software al definir las clases principales que componen la aplicación, así como sus atributos y métodos, además de las relaciones existentes entre ellas.

La estructura del presente documento se ha realizado de acuerdo al Software Engineering Standards Committee of the IEEE Computer Society; standard IEEE Std 830-1998, junto con las directrices del método Orientado a Objetos propuestas por Larman [Larman, 1999].

#### **2.1.1 Propósito**

El propósito de esta sección es definir el conjunto de Especificaciones de Requisitos Software que se deberán cumplir, tanto en la parte cliente de la aplicación o Addin, como en la parte del servidor de servicios web o Web Service, los cuales permiten definir el conjunto de pasos necesarios para que el Jefe de Proyectos pueda obtener la información o conocimiento que solicita, en función de las características de las planificaciones que gestiona.

A partir de la especificación de requisitos estaremos en condiciones de establecer un diseño que se ajuste a los requerimientos expuestos, después de realizar el estudio de los componentes necesarios.

#### **2.1.1.1 Audiencia a la que va dirigido**

Se dirige a todos aquellos interesados en la búsqueda constante de mejoras aplicables a los procesos de planificación y gestión de proyectos software.

#### **2.1.2 Alcance**

Cuando un Jefe de Proyecto realiza la planificación o seguimiento de un proyecto software, usando las herramientas y técnicas que considere más adecuadas, pueden surgir dudas con respecto a cómo las tareas y/o actividades afectan al proyecto por no tener conocimiento suficiente (tareas principales, sub-tareas, enlaces, duraciones, costes, recursos, etc.). Esta situación puede dar lugar a no cumplir con el plan del proyecto o, en el peor de los casos, a su fracaso.

Una posible ayuda para este tipo de situaciones se puede obtener del uso de artefactos de conocimiento, denominados “*Patrones de Producto*”, los cuales aportan pautas para la aplicación de las Buenas Prácticas de la Ingeniería del Software. Estos Patrones de Producto están basados en la experiencia de los ingenieros software expertos y en el conocimiento de proyectos previos, cuya información ha sido estructurada, dispuesta en bases de datos y publicadas por medio de una Wiki (ver definición en el apartado 2.1.1) para que pueda ser utilizada en el desarrollo de proyectos software.

De este modo, se relacionan ambas naturalezas, la de la planificación establecida por el Jefe de Proyecto junto con la del conocimiento reutilizable, obteniendo un nivel de información más minucioso, detallado y preciso, con respecto a las tareas referentes a la planificación, a través de los Patrones de Producto.

### 2.1.2.1 Funcionalidades

La aplicación será capaz de consultar la base del conocimiento, implementada mediante una Wiki y constituida dentro de sistemas de almacenamiento creados para tal efecto. Combinando restricciones y contextos junto con las tareas pertenecientes a las planificaciones se ofrecen los “*Patrones de Producto*” que cumplen con los requisitos del proyecto que pueden proporcionar conocimiento, los cuales serán asociados a las tareas seleccionadas, dentro de las mismas planificaciones, bajo petición y discriminación de los Jefes de Proyecto.

La aplicación permitirá:

- Listar y seleccionar el contexto (descripción funcional del sistema, especificación de requisitos definido, etc.), las restricciones genéricas (de organización: PYMES, grandes empresas, etc.) y las restricciones específicas (de proyecto: descripción estático del sistema, etc.) para las cuales se desea realizar la búsqueda de Patrones de Producto.
- Listar y seleccionar la tarea sobre la cual se pretende realizar la consulta de Patrones de Producto, combinada con el contexto y restricciones seleccionadas previamente.
- Crear y/o seleccionar aquellas columnas donde se quiera registrar la información dentro del propio cronograma de MS-Project.
- Consultar y/o seleccionar toda la información referente a los Patrones de Producto obtenidos tras la consulta a la base de conocimiento.
- Listar y seleccionar aquellas tareas que vayan a ser objeto de actualización por parte del patrón seleccionado.

Para conseguir llevar a cabo estas funcionalidades, la aplicación debe ser capaz de:

- Embeberse dentro del software MS-Project, desde la que puede ser llamada desde múltiples accesos (Opciones de menú y opciones de herramientas)

- Tener acceso al descriptor WSDL (Web Services Description Language) donde se muestran, públicamente, los servicios ofertados y que pueden ser consumidos por todas las llamadas entrantes, independientemente del lugar desde donde se realicen, siempre y cuando cumpla con el requisito de validación de usuarios.
- A su vez, el Web Service tendrá acceso a los servidores de bases de datos necesarios donde se aloja la información almacenada para realizar tanto las consultas de patrones como la inserción de los parámetros del proyecto actualizado.

#### **2.1.2.2 Beneficios, objetivos y metas**

Por todo lo anteriormente expuesto, queda clara la idea de que el presente proyecto tiene como objetivo ayudar a los Jefes de Proyecto, en base a conocimiento reutilizable, analizado y estructurado previamente, a tener una información más completa y detallada de la planificación de sus proyectos, de modo que la calidad de estos, así como su porcentaje de éxito, sea lo más alto posible.

## **2.2 DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS**

### **2.2.1 Definiciones**

#### **Actividad**

Expresa un trabajo a realizar para alcanzar un resultado específico. La actividad requiere tiempo y recursos. Puede comprender un conjunto de acciones y tareas. [Sanchez-ISW2, 2007]

#### **Addin**

Son programas anexados a otros que sirven para incrementar y/o complementar sus funcionalidades.

**Archivo o Librería DLL (Dynamic-Link Library)**

Es el término con el que se refiere a los archivos con código ejecutable que se cargan bajo demanda de un programa por parte del sistema operativo o de una aplicación determinada. [INET, 25]

Otra definición válida es que se trata de un archivo ejecutable que permite a los programas compartir código y otros recursos necesarios para realizar determinadas tareas. [INET, 26]

**Cliente de Web Service**

Realiza el consumo de los servicios publicados en el descriptor WSDL y recibe el resultado producido por estos, procesándolo de manera independiente. El cliente puede realizar sus peticiones desde cualquier lugar, siempre que tenga acceso a la URL donde se halle el descriptor WSDL.

**Descriptor WSDL**

Permite describir la interfaz pública de los servicios web (sus métodos y atributos) mediante XML. Detalla los protocolos y los formatos de los mensajes necesarios para interactuar con los servicios listados en su catálogo. [INET, 28]

**Estado de la Tarea**

Puede tomar diferentes valores según el momento en el que se encuentre: no comenzada, en proceso, parada, abandonada o terminada. [Sanchez-ISW2, 2007]

**Gestión de Proyectos**

Conjunto de procesos, técnicas y sistemas para planificar, de forma eficaz, un proyecto. Implica manejar, interrelacionar y asignar tareas y recursos (personal, equipamiento y materiales). [Recio-GO, 2007]

**Jefe de Proyecto o JP**

Usuario que tiene los permisos y conocimientos necesarios para planificar, gestionar, controlar y realizar el seguimiento de proyectos, descompuesto en tareas y sub-tareas, con



la posibilidad de existir dependencias entre ellas, y con el objetivo de obtener el éxito del proyecto, minimizando tiempos, costes y recursos.

Es recomendable que el Jefe de Proyecto posea habilidades personales, habilidades de gestión y habilidades técnicas para realizar, con más garantías, sus cometidos. [Recio-GO, 2007]

### **Patrones de Producto o Product Patterns**

Son artefactos de conocimiento estructurado que permiten encapsular y reutilizar el conocimiento de proyectos software previos.

### **PMBOK**

Es una colección de procesos y áreas de conocimiento generalmente aceptadas como las mejores prácticas dentro de la gestión de proyectos. El PMBOK es un estándar reconocido internacionalmente (IEEE Std 1490-2003) que provee los fundamentos de la gestión de proyectos que son aplicables a un amplio rango de proyectos, incluyendo construcción, software, ingeniería, etc. [INET, 30]

### **Proyecto**

Conjunto de actividades, eventos e hitos entrelazados para la consecución de un resultado común. [Sanchez-ISW2, 2007] Se trata de una iniciativa que cuenta con un principio y un final, llevado a cabo por personas, para alcanzar un objetivo específico y dentro de unos parámetros acordados de tiempo, coste y calidad. [Recio-GO, 2007]

### **Servicio Web o Web Services**

Es una aplicación software identificada mediante una URL, normalmente usando el protocolo SOAP, cuyo interfaz (y uso) es capaz de ser definido, descrito y descubierto mediante artefactos XML, y soportar interacciones directas con otras aplicaciones software usando mensajes basados en XML y protocolos basados en Internet. [INET, 27]

Recibe la solicitud de los clientes, la procesa y devuelve una respuesta en virtud de los parámetros de entrada.

**Servidor GlassFish**

Es un servidor de aplicaciones, gratuito y de código libre, desarrollado por Sun Microsystems que implementa las tecnologías definidas en la plataforma JavaEE y permite ejecutar aplicaciones empresariales que siguen esta especificación (transaccionalidad, aplicaciones distribuidas, arquitecturas MOM, desarrollo portales WEB, desarrollo SOA, etc.). [INET, 31]

**Tarea**

Actividad que deriva en un resultado entregable único. Supone el nivel de detalle mínimo. [Sanchez-ISW2, 2007]

**Técnica/s**

Es un procedimiento o conjunto de estos, (reglas, normas o protocolos), que tienen como objetivo obtener un resultado determinado, ya sea en el campo de la ciencia, de la tecnología, del arte, de la educación o en cualquier otra actividad. [INET, 29].

**Wiki**

Un wiki, o una wiki, es un sitio web cuyas páginas web pueden ser editadas por múltiples voluntarios a través del navegador web. Los usuarios pueden crear, modificar o borrar un mismo texto que comparten. Los textos o «páginas wiki» tienen títulos únicos. Si se escribe el título de una «página wiki» en algún lugar del wiki, esta palabra se convierte en un «enlace web» a la página web. [INET, 36]

**2.2.2 Abreviaturas**

<b>COCOMO</b>	COConstructive COst MOdel
<b>GNU</b>	GNU's Not Unix
<b>MSProject</b>	Microsoft Project
<b>MSSQL</b>	Microsoft SQL Server
<b>SOFTCOST</b>	Software Cost Estimation Process
<b>SPQR</b>	Software Productivity Quality and Reliability
<b>SW</b>	Software

### 2.2.1 Acrónimos

<b>AON</b>	Activity on a Node
<b>CMM</b>	Capability Maturity Model
<b>CPM</b>	Critical Path Method
<b>CRM</b>	Customer Relationship Management
<b>DLL</b>	Dynamic Link Library
<b>GPL</b>	General Public License
<b>GNOME</b>	GNU Network Object Model Environment
<b>JP</b>	Jefe/s de Proyecto/s
<b>KDE</b>	K Desktop Environment
<b>MOM</b>	Message Oriented Middleware
<b>MSP</b>	Microsoft Project
<b>PC</b>	Personal Computer
<b>PDM</b>	Precedence Diagram Method
<b>PERT</b>	Program Evaluation and Review Technique
<b>PFC</b>	Proyecto Fin de Carrera
<b>PHP</b>	Hypertext Pre-processor
<b>PMBOK</b>	Project Management Body of Knowledge
<b>PPTS</b>	Project Planning and Tracking System
<b>PMT</b>	Project Management Institute
<b>RBS</b>	Risk Breakdown Structure
<b>SMC</b>	Supply Chain Management
<b>SOAP</b>	Simple Object Access Protocol
<b>UDDI</b>	Universal Description, Discovery, and Integration
<b>URL</b>	Uniform Resource Locator
<b>VB</b>	Visual Basic
<b>WBS</b>	Work Breakdown Structure
<b>WSDL</b>	Web Services Description Language
<b>XML</b>	Extensible Markup Language

### **2.2.2 Apreciación Global**

Primeramente, se realizará una descripción general del producto que se va a desarrollar, para, posteriormente, pasar a estudiar cada uno de los requisitos específicos, haciendo referencia particular a los casos de uso que ofrece la metodología propuesta por Craig Larman [Larman, 1999].

## **2.3 DESCRIPCIÓN GLOBAL**

### **2.3.1 Perspectiva del Producto.**

La aplicación consta de tres piezas, claramente identificables por su funcionalidad: el cliente, el Servidor de Servicios y, por último, el gestor de bases de datos, el cual se encarga de almacenar todo el conocimiento consultado.

La parte cliente (o componente Addin) se encuentra embebida dentro del programa Microsoft Project [Chatfield, 2004] de Office, y se instala en los equipos clientes por medio de un sencillo auto-instalable, ligeramente configurable, y de creación ajena a la aplicación.

Por otro lado, el Servidor de Servicios (o Web Service) [Chappell, 2002] funciona gracias a un servidor de aplicaciones web y ambos se alojan en un equipo servidor destinado para tal uso. Gracias al descriptor WSDL del Web Service, y a través de una dirección web URL, se hacen accesibles y públicos los servicios a los clientes.

Por último, se encuentra el gestor de base de datos, el cual se aloja en el equipo servidor, de igual manera que el Web Service y el servidor de servicios web. Este gestor de base de datos se encarga de tramitar las peticiones de consulta, realizadas por parte de los clientes, para obtener la información necesaria que aporte conocimiento a las planificaciones realizadas por los Jefes de Proyectos, así como para guardar históricos de modificaciones ejecutados sobre estos calendarios.

### 2.3.2 Funciones del Producto

En esta sección se muestra un listado de funcionalidades ofrecidas por la herramienta software con el objetivo de ayudar a los Jefes de Proyecto a aportar conocimiento a los procesos de planificación de proyectos que gestionan. Para ello, el Jefe de Proyecto debe realizar una serie de pasos secuenciados que sirven para obtener los patrones de producto, los cuales pueden ser aplicables a la planificación, según el caso y las características de la consulta, tanto en información detallada, en la aplicación de Buenas Prácticas, como en recomendaciones sobre tiempos estimados para la realización de las tareas para las cuales se utilizó la herramienta en la búsqueda de conocimiento.

El listado de funcionalidades, asignadas a la herramienta software, definidas para obtener el resultado deseado, consta de los siguientes pasos:

- Definición parcial o total de la planificación del proyecto que se pretende realizar, es decir, la herramienta puede ser usada sin necesidad de tener finalizada la etapa de planificación del proyecto por lo que se puede aportar conocimiento a una tarea de manera independiente al resto.
- Llamada de la herramienta software que da soporte y ayuda al Jefe de Proyecto de la planificación del proyecto.
- Selección del contexto sobre el que se pretende realizar la búsqueda (descripción funcional del sistema, especificación de requisitos definida, etc.).
- Selección de la restricción genérica (de organización: PYMES, grandes empresas, etc.).
- Selección de la restricción específica (de proyecto: descripción estática del sistema, etc.), dependiente de la restricción genérica.
- Selección de la tarea sobre la que se pretende realizar la consulta de mejora en su definición y detalle.
- Realización de la consulta y obtención del listado de patrones relacionados, según las características de búsqueda.

- Creación de la columna dentro de MSProject donde se almacena la dirección URL (la cual accede a la información detallada que está publicada en la Wiki), que contiene la información referente al patrón seleccionado.
- Selección de las tareas de la planificación del proyecto que serán objeto de la asignación del conocimiento reutilizable seleccionado.

La base de datos utilizada ocupa una posición vital dentro de la funcionalidad de la herramienta, ya que almacena toda la información y el conocimiento, previamente estudiado, analizado y estructurado, el cual también es accesible a través de una Wiki. Este conocimiento permite ayudar a los Jefes de Proyecto a proporcionar al equipo de desarrollo las Buenas Prácticas de los Ingenieros de Software expertos y a estimar el tiempo de implementación de las tareas, ya que los Patrones de Producto tienen asociado, además de una guía de los pasos a realizar para obtener los productos software de cada tarea, los plazos de tiempos para completarlos.

Las funcionalidades que involucran a la base de datos, a través del Web Service son las siguientes:

- Consulta sobre el listado de Contextos existentes.
- Consulta sobre el listado de Restricciones Genéricas existentes.
- Consulta sobre el listado de Restricciones Específicas existentes dependiendo de la Restricción Genérica de la que dependen y que fue seleccionada previamente.
- Consulta sobre el listado de Patrones de Producto existentes dependiendo de los parámetros de consulta seleccionados previamente (contexto, restricciones y tarea).
- Guarda todas las características relevantes del proyecto (tareas y los patrones seleccionados, perfiles, etc.) para el cual se ha realizado y finalizado con éxito la búsqueda y asignación de los Patrones de Producto, de modo que sirva como un histórico de versiones.

### 2.3.3 Requisitos Funcionales

A continuación, se describen los requisitos que ha de cumplir la aplicación para alcanzar con éxito las funcionalidades deseadas:

<b>Req(01)</b>	El sistema debe permitir ser usado solo por usuarios registrados.
<b>Req(02)</b>	El usuario podrá consultar la validez del enlace del fichero WSDL.
<b>Req(03)</b>	El sistema debe ser capaz de recuperar de la base de datos todos los contextos existentes.
<b>Req(04)</b>	El sistema debe ser capaz de recuperar de la base de datos todas las restricciones genéricas existentes.
<b>Req(05)</b>	El sistema debe ser capaz de recuperar de la base de datos todas las restricciones específicas existentes en función de una restricción genérica dada.
<b>Req(06)</b>	El sistema debe ser capaz de mostrar el listado de todas las tareas de las que se compone el fichero de planificación en el que se está ejecutando la aplicación.
<b>Req(07)</b>	El sistema debe mostrar el estado en el que se encuentra durante el proceso de búsqueda de patrones de producto.
<b>Req(08)</b>	El sistema debe ser capaz de recuperar de la base de datos todos los patrones existentes relacionados con una búsqueda dada.
<b>Req(09)</b>	El sistema debe ser capaz de recuperar de la base de datos la información relativa al patrón seleccionado.
<b>Req(10)</b>	El sistema debe permitir a los usuarios buscar patrones según el contexto y la restricción específica seleccionados.
<b>Req(11)</b>	El sistema debe permitir elegir una tarea como origen de la búsqueda de patrones.
<b>Req(12)</b>	El sistema debe permitir crear la columna destino que albergará como valor el patrón seleccionado.
<b>Req(13)</b>	El sistema debe permitir seleccionar una, varias o todas las tareas del calendario como destinatarios para la asignación del patrón seleccionado.
<b>Req(14)</b>	El sistema debe permitir deshacer la selección de una, varias o todas las tareas del calendario como destinatarios para la asignación del patrón seleccionado.

<b>Req(15)</b>	El sistema debe poder acceder a la base del conocimiento que se encuentra implementada en una Wiki publicada en internet para obtener la información del patrón seleccionado.
<b>Req(16)</b>	El sistema debe mostrar mensajes del estado en que se encuentre en cada operación, lo más claros posibles.
<b>Req(17)</b>	El sistema debe dar información clara y aportar soluciones constructivas ante los posibles errores que se produzcan.
<b>Req(18)</b>	En el momento en el que el usuario confirme la operación, el sistema deberá actualizar la información de las tareas destino del MS-Project (por medio del campo creado para dicho fin en el calendario de planificación) con el enlace que accede al Patrón de Producto, seleccionado durante el proceso de consulta del conocimiento.
<b>Req(19)</b>	El usuario podrá cerrar por completo la aplicación en cualquier momento, siendo avisado de que se perderán todos los cambios realizados que no fueran guardados con anterioridad.
<b>Req(20)</b>	Habrà una opción de salir, para las pantallas secundarias, con la que se abandonarán las mismas sin que con ello se cierre la aplicación completamente.
<b>Req(21)</b>	El usuario, una vez abierto un documento MS-Project, solo necesitará introducir su identificación la primera vez que acceda al sistema para dicho documento.
<b>Req(22)</b>	Habrà una opción de salir durante el asistente de creación de la columna destino de la actualización de la información de la tarea, con la que se abandonará dicho proceso del asistente.
<b>Req(23)</b>	La aplicación debe mostrar una ayuda adicional, al situar el puntero del ratón sobre algún elemento gráfico de la aplicación, para informar al usuario de la finalidad del elemento sobre el que se encuentra (mediante un sistema conocido por el nombre de <i>“ToolTips”</i> ).
<b>Req(24)</b>	El sistema debe permitir la inserción de la información del proyecto para su posterior recuperación.



### 2.3.4 Requisitos de Interfaz

#### 2.3.4.1 Interfaz con el Usuario.

Las salidas de la herramienta van destinadas al usuario del mismo (Jefes del Proyecto), y serán ofrecidas a través de la pantalla.

Para comunicarse con el ordenador, el Jefe de Proyecto utilizará el ratón y el teclado.

### 2.3.5 Requisitos No Funcionales.

#### 2.3.5.1 Requisitos Hardware.

En cuanto a hardware se refiere, con el fin de que la herramienta sea lo más accesible posible, basta con un ordenador personal estándar del mercado (PC), tanto para el servidor como para el cliente.

Los requisitos no funcionales que se han de cumplir son:

<b>Req-NF(01)</b>	Fácil de aprender.
<b>Req-NF(02)</b>	Fácil de usar.
<b>Req-NF(03)</b>	Uso fácil de recordar.
<b>Req-NF(04)</b>	Navegación sencilla.
<b>Req-NF(05)</b>	Colores adecuados.
<b>Req-NF(06)</b>	Usable.
<b>Req-NF(07)</b>	Entorno gráfico agradable.
<b>Req-NF(08)</b>	El sistema debe cumplir las disposiciones recogidas en la Ley Orgánica de Datos Personales y en el Reglamento de medidas de seguridad.
<b>Req-NF(09)</b>	El sistema debe tardar menos de 5 segundos en mostrar los resultados de la búsqueda de patrones búsqueda.
<b>Req-NF(10)</b>	El servidor de servicios debe ser montado sobre el sistema operativo Linux Ubuntu.

<b>Req-NF(11)</b>	La base de datos que contiene toda la información de los patrones debe estar montada sobre el sistema operativo Linux Ubuntu.
<b>Req-NF(12)</b>	El sistema debe funcionar sobre MS-Project, versión igual o superior a la del 2003, el cual debe estar instalado en los equipos clientes sobre un sistema operativo Windows, correctamente configurado y con unos requisitos mínimos hardware que permitan soportar su ejecución.

#### **2.3.5.2 Requisitos Software.**

La aplicación deberá funcionar en ordenadores con sistema operativo que soporten Microsoft Project 2003, con sistema operativo Windows XP, Windows Vista o Windows 7, además de tener disponible un navegador capaz de soportar el manejo de ficheros XML.

## 2.4 CASOS DE USO

### 2.4.1 Definición de los Casos de Uso.

Según definió Jacobson [Jacobson, 1992], un “*Caso de Uso*” es un documento narrativo que describe la secuencia de eventos de un actor (agente externo) que usa un sistema para completar un proceso, es decir, una forma de usar la función que ofrece el sistema. [Ferré, UML]

Los Casos de Uso [Jacobson, 1992] describen, bajo la forma de acciones y reacciones, el comportamiento de un sistema desde el punto de vista del usuario, permiten definir los límites del sistema y las relaciones entre el sistema y el entorno. Por lo tanto, actúan como descripciones de la funcionalidad del sistema, independientes de la implementación. [INET, 33]

Para la definición de los casos “Casos de Uso de Alto Nivel” y en “Casos de Uso en Formato Expandido”, la metodología de Craig Larman [Larman, 1999] dedica varios capítulos definidos bajo el apartado dedicado a la “Fase de Planificación y Especificación de Requisitos”, los cuales han sido seguidos con objeto de su aplicación en las siguientes secciones.

### 2.4.2 Diagrama de casos de uso.

En esta sección se muestra el Diagrama de Casos de Uso, el cual permite visualizar el conjunto de cada uno de ellos, el actor y las relaciones existentes, identificados dentro de la herramienta software. Este tipo de diagramas nos permite capturar información sobre cómo trabaja nuestra herramienta, orientada a la captura de requisitos.

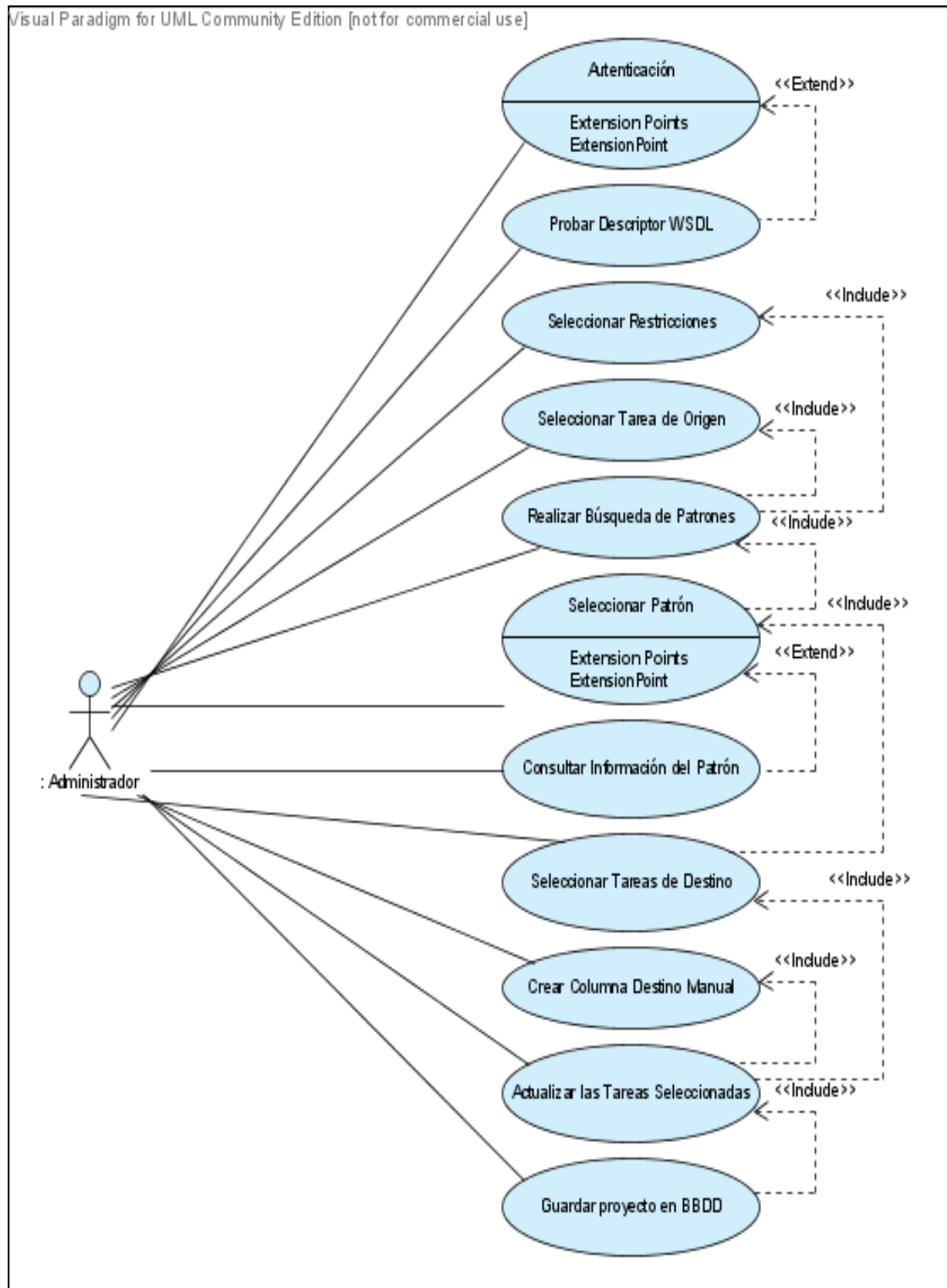


Figura 18. Diagrama de Casos de Uso de la Aplicación

### 2.4.3 Casos de Uso de Alto Nivel

Esta sección nos ofrece una visión abstracta de la funcionalidad de la herramienta mediante la descripción de los “Casos de Uso de Alto Nivel”, lo cual nos permite tener una idea general y resumida de lo que se puede realizar con la herramienta.

#### 2.4.3.1 Entrada al sistema

**Caso de Uso:** Autenticación

**Actor/es:** Jefe de Proyecto

**Tipo:** Primario

**Descripción:** El sistema le pedirá al Jefe de Proyecto la dirección URL del *Descriptor WSDL*, el nombre de *Usuario* y *Contraseña* con el que desea acceder a la aplicación. El sistema comprobará que los valores introducidos son correctos probando la conexión con la Base de Datos.

**Caso de Uso:** Probar Descriptor WSDL

**Actor/es:** Jefe de Proyecto

**Tipo:** Primario

**Descripción:** El Jefe de Proyecto verifica que la dirección *URL* introducida como parámetro corresponde adecuadamente a la dirección del *Descriptor WSDL*, el cual ofrece los servicios necesarios que dan soporte a nuestra aplicación. Para ello, hace uso del servicio “*Probar WSDL*” del descriptor WSDL, el cual debe devolver el valor “Verdadero” en el caso de que no haya existido ningún problema a la hora de establecer la comunicación

#### 2.4.3.2 Selección de Parámetros de Búsqueda

**Caso de Uso:** Seleccionar Parámetros

**Actor/es:** Jefe de Proyecto

**Tipo:** Primario

**Descripción:** El Jefe de Proyecto selecciona un *Contexto*, una *Restricción Genérica* y una *Tarea de Origen*. En virtud de esta última selección, se genera un listado de *Restricciones Específicas* del cual debe seleccionar, nuevamente, una de ellas. El *Contexto*, la *Restricción Específica* y la *Tarea de Origen* forman parte de los parámetros necesarios para realizar la consulta de Patrones de Producto que se llevará a cabo sobre la base de datos.

**Caso de Uso:** Realizar Búsqueda de Patrones

**Actor/es:** Jefe de Proyecto

**Tipo:** Primario

**Descripción:** El Jefe de Proyecto selecciona la opción para comenzar la *Búsqueda de Patrones* y los resultados se obtendrán de la base de datos dependiendo del Contexto, Restricciones y Tarea de origen seleccionados previamente.

#### 2.4.3.3 Asignación de Patrones

**Caso de Uso:** Seleccionar Patrón

**Actor/es:** Jefe de Proyecto

**Tipo:** Primario

**Descripción:** El Jefe de Proyecto selecciona un *Patrón de Producto* del conjunto de Patrones de Producto que se le presentan, como resultado de la búsqueda realizada previamente.

**Caso de Uso:** Consultar Información del Patrón

**Actor/es:** Jefe de Proyecto

**Tipo:** Primario

**Descripción:** El Jefe de Proyecto selecciona el *enlace URL* presentado tras la selección de uno de los Patrones de Producto (véase Caso de Uso *Seleccionar Patrón*) de entre el listado de coincidentes, en función de los parámetros de búsqueda. A continuación, el navegador, configurado por defecto, se conectará a dicho enlace a través de Internet y le mostrará la información detallada contenida en ella.

#### 2.4.3.4 Actualización de Tareas

**Caso de Uso:** Seleccionar Tareas de Destino

**Actor/es:** Jefe de Proyecto

**Tipo:** Primario

**Descripción:** El Jefe de Proyecto selecciona una o varias de las Tareas existentes en el proyecto actual, las cuales serán objeto de destino para la actualización del Patrón de Producto seleccionado previamente (véase Caso de Uso *Seleccionar Patrón*). El Jefe de Proyecto dispone de la opción de seleccionar o deshacer la selección de todas las tareas. Posteriormente, se actualiza el contenido de la columna creada (véase Caso de Uso *Crear Columna Destino*) con el fin de albergar la dirección URL de dicho Patrón de Producto.

**Caso de Uso:** Crear Columna Destino

**Actor/es:** Jefe de Proyecto

**Tipo:** Primario

**Descripción:** El Jefe de Proyecto selecciona la opción de crear la columna de destino objeto de la actualización de las tareas. Para ello, se le muestra una pequeña guía-asistente para realizar dicha operación.

**Caso de Uso:** Guardar Proyecto en la BBDD

**Actor/es:** Jefe de Proyecto

**Tipo:** Primario

**Descripción:** El Jefe de Proyecto hace uso de la opción de guardado del proyecto actual en la base de datos, con intención de servir como histórico de versiones, almacenando las nuevas asignaciones realizadas pudiendo ser recuperadas cuando así se considere necesario.

### 2.4.4 Casos de Uso en Formato Expandido

Esta sección nos ofrece una visión más completa y menos abstracta que la sección anterior (2.4.3) sobre la funcionalidad de la herramienta, mediante la descripción de los “Casos de Uso en Formato Expandido”.

#### 2.4.4.1 Caso de Uso: Autenticación

**Actor/es:** Jefe de Proyecto (iniciador)

**Propósito:** Entrar a la aplicación

**Tipo:** Primario y esencial

**Visión General:** El sistema le pedirá al Jefe de Proyecto la dirección URL del *Descriptor WSDL*, el nombre de *Usuario* y *Contraseña* con el que desea acceder a la aplicación. El sistema comprobará que los valores introducidos son correctos probando la conexión con la Base de Datos.

**Referencias:** Req(01), Req(16), Req(21)




Curso típico de eventos:



Usuario	Sistema
1 Accede a la aplicación.	2 Se solicita una dirección URL para el descriptor WSDL, un nombre de usuario y una contraseña para poder acceder.
3 Indica la URL del descriptor WSDL, el usuario y la contraseña.	4 Establece la conexión con la ruta URL del descriptor WSDL.



	<p>5 Llama al servicio “connectionRequest”, proporcionado por el Web Service, al que se le pasan los parámetros Usuario/Contraseña introducidos previamente. Este servicio comprueba que el usuario existe en la Base de Datos.</p> <p>6 Muestra el asistente para crear la columna sobre el calendario que será destino de la información del patrón seleccionado más adelante. (véase el Caso de Uso en Formato Expandido <i>Crear Columna Destino</i>)</p> <p>7 Permite al usuario acceder a la aplicación.</p>
8 Sigue el asistente, confirma los pasos y crea la columna destino.	9 Crea la columna destino con las características definidas por el usuario.

## Cursos alternativos:

-  Paso 4: No hay conexión con el descriptor WSDL. Se muestra un mensaje de error y cancela la operación.
-  Paso 5: El servicio no responde. Se muestra un mensaje de error y cancela la operación.
-  Paso 5: No hay conexión con la Base de Datos. Se muestra un mensaje de error y cancela la operación.

-  Paso 5: La combinación Usuario/Contraseña introducidos no tiene acceso a la Base de Datos. Se muestra un mensaje de error y vuelve a pedir los datos.
-  Paso 8: El usuario cancela el asistente para la creación de la columna destino objeto de la actualización.

#### 2.4.4.2 Caso de Uso: Probar Descriptor WSDL

**Actor/es:** Jefe de Proyecto (iniciador)

**Propósito:** Permite al usuario probar si la ruta URL del Descriptor WSDL es válida.

**Tipo:** Primario y esencial

**Visión General:** El Jefe de Proyecto verifica que la dirección *URL* introducida como parámetro corresponde adecuadamente a la dirección del *Descriptor WSDL*, el cual ofrece los servicios necesarios que dan soporte a nuestra aplicación. Para ello, hace uso del servicio “*Probar WSDL*” del descriptor WSDL, el cual debe devolver el valor “Verdadero” en el caso de que no haya existido ningún problema a la hora de establecer la comunicación.



**Referencias:** Req(02), Req(16)

Curso típico de eventos:

Usuario	Sistema
1 Accede a la aplicación.	2 Se solicita una dirección URL para el descriptor WSDL para poder verificarla.
3 Indica la URL del descriptor WSDL.	4 Establece la conexión con la ruta URL del descriptor WSDL.

	<p>5 Llama al servicio “TestWSDL”, proporcionado por el Web Service, el cual devolverá la confirmación del establecimiento de la conexión, validando la dirección URL introducida previamente.</p> <p>6 El resultado de la operación se muestra por medio de un mensaje emergente.</p>
--	--

Cursos alternativos:

-  Paso 4: No hay conexión con el descriptor WSDL. Se muestra un mensaje de error y vuelve a pedir los datos.
-  Paso 5: El servicio no responde. Se muestra un mensaje de error.

#### 2.4.4.3 Caso de Uso: Seleccionar Parámetros

**Actor/es:** Jefe de Proyecto (iniciador)

**Propósito:** Seleccionar los parámetros necesarios para realizar la búsqueda de los Patrones.

**Tipo:** Primario y esencial







**Visión General:** El Jefe de Proyecto selecciona un *Contexto*, una *Restricción Genérica* y una *Tarea de Origen*. En virtud de esta última selección, se genera un listado de *Restricciones Específicas* del cual debe seleccionar, nuevamente, una de ellas. El *Contexto*, la *Restricción Específica* y la *Tarea de Origen* forman parte de los parámetros necesarios para realizar la consulta de Patrones de Producto que se llevará a cabo sobre la base de datos.

**Referencias:** Req(01), Req(03), Req(04), Req(05), Req(06), Req(11)

Curso típico de eventos:

<b>Usuario</b>	<b>Sistema</b>
1 Accede a la pantalla de selección de los parámetros de búsqueda: Contextos y Restricciones.	2 Muestra un listado con todos los Contextos y Restricciones Genéricas existentes en la Base de Datos.  3 Muestra un listado con todas las Tareas existentes en el proyecto actual.
4 Selecciona uno de los Contextos.  5 Selecciona una de las Restricciones Genéricas.	6 Muestra otro listado con todas las Restricciones Específicas existentes en la Base de Datos, dependiendo de la Restricción Genérica seleccionada previamente.
7 Selecciona una de las Restricciones Específicas.	8 Habilita la opción para almacenar temporalmente el valor de las opciones seleccionadas.
9 Selecciona una de las Tareas existentes en el proyecto como Tarea Origen de la búsqueda.	10 Habilita la opción para comenzar la búsqueda de Patrones.

Cursos alternativos:

-  Paso 2: No existe ningún listado de Contextos en la Base de Datos que pueda ser mostrada. Se deshabilita la opción para continuar con la operación.
-  Paso 2: No existe ningún listado de Restricciones Genéricas en la Base de Datos que pueda ser mostrada. Se deshabilita la opción para continuar con la operación.
-  Paso 3: El proyecto no tiene ninguna Tarea definida aun que pueda ser mostrada. Se deshabilita la opción comenzar la búsqueda.
-  Paso 6: No existe ningún listado de Restricciones Específicas en la Base de Datos que pueda ser mostrada. Se deshabilita la opción para continuar con la operación.
-  Paso 8: Alguno de los parámetros necesarios para realizar la búsqueda no están seleccionados (Contexto, Restricción Genérica y Restricción Específica). Se deshabilita la opción para continuar con la operación.
-  Paso 9: Alguno de los parámetros necesarios para realizar la búsqueda no están seleccionados (Contexto, Restricción Genérica, Restricción Específica y Tarea Origen). Se deshabilita la opción comenzar la búsqueda.

#### 2.4.4.4 Caso de Uso: Realizar Búsqueda de Patrones

**Actor/es:** Jefe de Proyecto (iniciador)

**Propósito:** Obtener un listado de Patrones en función de unos parámetros de entrada.

**Tipo:** Primario y esencial

**Visión General:** El Jefe de Proyecto selecciona la opción para comenzar la *Búsqueda de Patrones* y los resultados se obtendrán de la base de datos dependiendo del Contexto, Restricciones y Tarea de origen seleccionados previamente.

**Referencias:** Req(01), Req(07), Req(08), Req(10), Req(16), Req(20)

Curso típico de eventos:

Usuario	Sistema
1 Selecciona la acción de buscar Patrones.	<p>2 Realiza la búsqueda de patrones, en función de los parámetros de entrada (Contexto, Restricción Específica y Tarea Origen de la búsqueda) obteniendo un listado de los coincidentes.</p> <p>3 Da acceso a la siguiente parte de la aplicación: Selección y asignación de Patrones.</p>

Cursos alternativos:



Paso 2: No existe ningún listado de Patrones en la Base de Datos que coincida con los parámetros de búsqueda introducidos.

#### 2.4.4.5 Caso de Uso: Seleccionar Patrón

**Actor/es:** Jefe de Proyecto (iniciador)

**Propósito:** Seleccionar uno de los Patrones, de entre los obtenidos en la búsqueda.

**Tipo:** Primario y esencial

**Visión General:** El Jefe de Proyecto selecciona un *Patrón de Producto* del conjunto de Patrones de Producto que se le presentan, como resultado de la búsqueda realizada previamente.

**Referencias:** Req(01), Req(08), Req(09)

Curso típico de eventos:

Usuario	Sistema
1 Accede a la pantalla de selección de Patrones.	2 Muestra un listado con todos los Patrones coincidentes como resultado de la búsqueda realizada previamente.
3 Selecciona uno de los Patrones.	4 Muestra la información relacionada con el Patrón seleccionado.  5 Se habilita la opción de asignación del Patrón a las Tareas del proyecto.

Cursos alternativos:



Paso 2: No se muestra ningún listado de Patrones dados los parámetros de búsqueda introducidos, por lo que no se muestra información adicional a ningún Patrón.

#### 2.4.4.6 Caso de Uso: Consultar Información del Patrón

**Actor/es:** Jefe de Proyecto (iniciador)

**Propósito:** Consultar información relativa al Patrón seleccionado.

**Tipo:** Primario y esencial

**Visión General:** El Jefe de Proyecto selecciona el *enlace URL* presentado tras la selección de uno de los Patrones de Producto (véase Caso de Uso *Seleccionar Patrón*) de entre el listado de coincidentes, en función de los parámetros de búsqueda. A continuación, el



navegador, configurado por defecto, se conectará a dicho enlace a través de Internet y le mostrará la información detallada contenida en ella.

**Referencias:** Req(01), Req(15)

Curso típico de eventos:

Usuario	Sistema
1 Selecciona uno de los Patrones.	2 Muestra la información relacionada con el Patrón seleccionado.  3 Se habilita la opción de asignación del Patrón a las Tareas del proyecto.
4 Selecciona la información URL del Patrón seleccionado.	5 Abre el navegador configurado por defecto para el usuario y carga la dirección URL del patrón seleccionado.

Cursos alternativos:

-  Paso 5: No existe un navegador por defecto y se solicita la selección de uno.
-  Paso 5: No existe un navegador instalado en el equipo. Se muestra un asistente para abrir con otro programa.



#### 2.4.4.7 Caso de Uso: Seleccionar Tareas de Destino

**Actor/es:** Jefe de Proyecto (iniciador)

**Propósito:** Seleccionar las Tareas a las que se les asignarán el Patrón seleccionado previamente y actualizar el contenido de la columna creada para almacenar la URL de dicho Patrón.

**Tipo:** Primario y esencial

**Visión General:** El Jefe de Proyecto selecciona una o varias de las Tareas existentes en el proyecto actual, las cuales serán objeto de destino para la actualización del Patrón de Producto seleccionado previamente (véase Caso de Uso *Seleccionar Patrón*). El Jefe de Proyecto dispone de la opción de seleccionar o deshacer la selección de todas las tareas. Posteriormente, se actualiza el contenido de la columna creada (véase Caso de Uso *Crear Columna Destino*) con el fin de albergar la dirección URL de dicho Patrón de Producto.




**Referencias:** Req(01), Req(06), Req(13), Req(14), Req(16), Req(18), Req(19), Req(20)

Curso típico de eventos:

Usuario	Sistema
1 Selecciona la opción para establecer el ámbito de la actualización de los Patrones.	2 Da acceso a la pantalla de selección de Tareas que serán destino de la actualización del patrón seleccionado.  3 Muestra un listado con todas las Tareas existentes en el proyecto actual.
4 Selecciona una o varias de las Tareas.	5. Habilita la opción de asignación / actualización.

6 Selecciona la acción de asignación / actualización de las Tareas.	7 Actualiza el campo creado en el proyecto, el cual almacena el valor de la URL del Patrón seleccionado.
---	--

Cursos alternativos:

-  Paso 3: El proyecto no tiene creada ninguna Tarea que pueda ser mostrada. Se deshabilita la opción de asignación.
-  Paso 4: El Jefe de Proyecto selecciona todas las Tareas de la lista mediante la opción “Seleccionar Todas”, por lo que se habilita la opción de asignación.
-  Paso 4: El Jefe de Proyecto deshace la selección de todas las Tareas de la lista mediante la opción “Deseleccionar Todas”, por lo que se deshabilita la opción de asignación.

#### 2.4.4.8 Caso de Uso: Crear Columna Destino

**Actor/es:** Jefe de Proyecto (iniciador)

**Propósito:** Crear una columna en el MSProject la cual será destino para la información del Patrón seleccionado.

**Tipo:** Primario y esencial

**Visión General:** El Jefe de Proyecto selecciona la opción de crear la columna de destino objeto de la actualización de las tareas. Para ello, se le muestra una pequeña guía-asistente para realizar dicha operación.

**Referencias:** Req(12), Req(16), Req(22)

Curso típico de eventos:

Usuario	Sistema
1 Selecciona la acción de crear una columna destino para la información de los Patrones seleccionados.	2. Muestra una pequeña guía con una serie de pautas y pasos recomendados a seguir.
3 Selecciona como <i>Nombre del Campo</i> el de “Texto 1” y un <i>Título</i> para la nueva columna. Finalmente, acepta.	4. Crea la columna destino con las características definidas por el usuario.

Cursos alternativos:



Paso 3: Incumple las indicaciones recomendadas, la información de los patrones seleccionados no será visible pero si existirá de manera oculta hasta que se muestre de manera manual.

#### 2.4.4.9 Caso de Uso: Guardar proyecto en BBDD

**Actor/es:** Usuario (iniciador)

**Propósito:** Copiar en la Base de Datos los datos más relevantes del proyecto a modo de mantener un histórico de versiones.

**Tipo:** Primario y esencial

**Visión General:** El Jefe de Proyecto hace uso de la opción de guardado del proyecto actual en la base de datos, con intención de servir como histórico de versiones, almacenando las nuevas asignaciones realizadas pudiendo ser recuperadas cuando así se considere necesario.

**Referencias:** Req(24)

Curso típico de eventos:

<b>Usuario</b>	<b>Sistema</b>
1. Selecciona la acción para guardar la información del proyecto actual.	2 Almacena, en la base de datos, la información mínima referente al proyecto: nombre del proyecto, fecha, organización, tareas y sus patrones asociados.

Cursos alternativos:

No hay

## 2.5 PLANIFICACIÓN

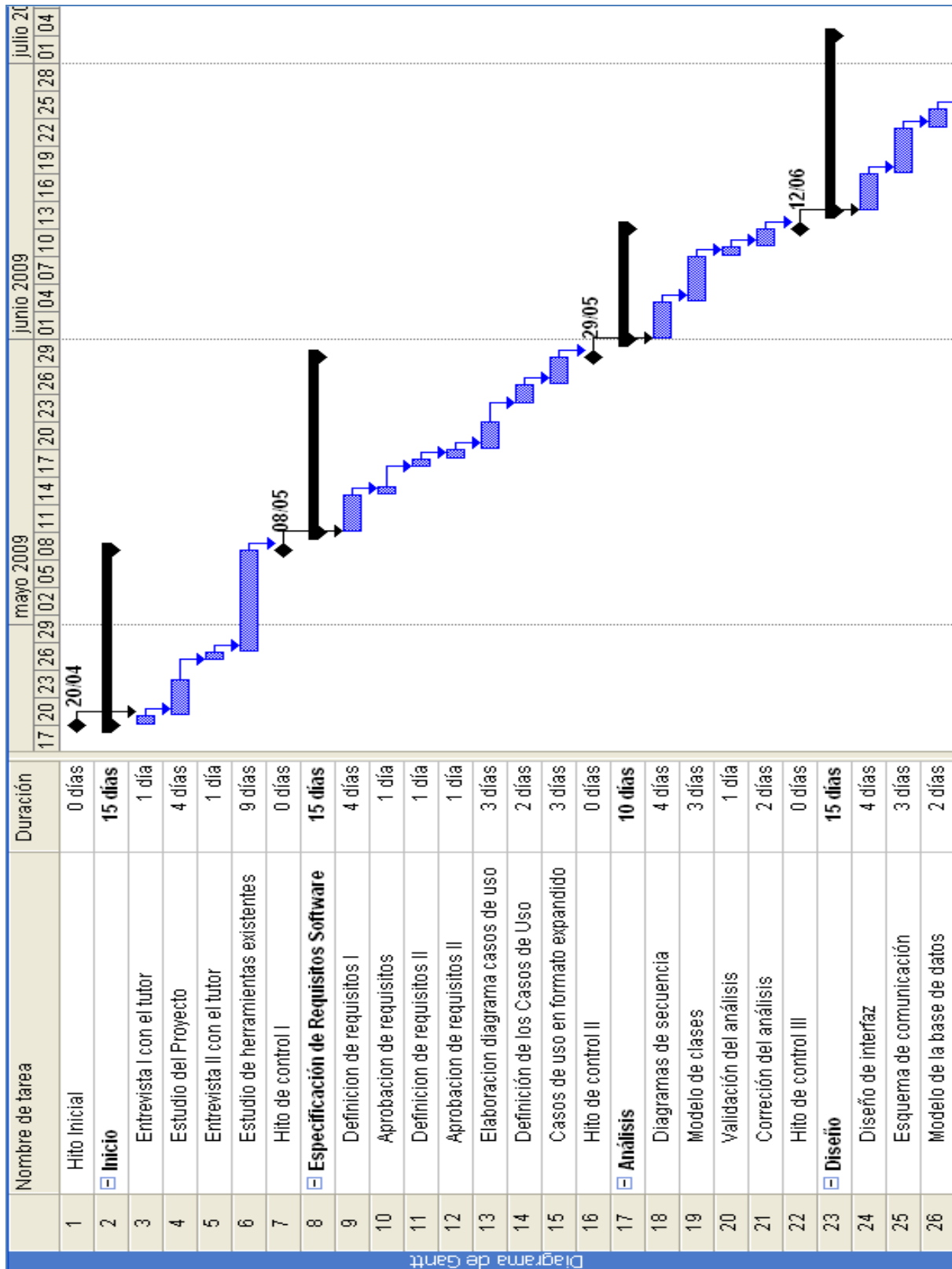


Figura 19a. Planificación del Proyecto (I)

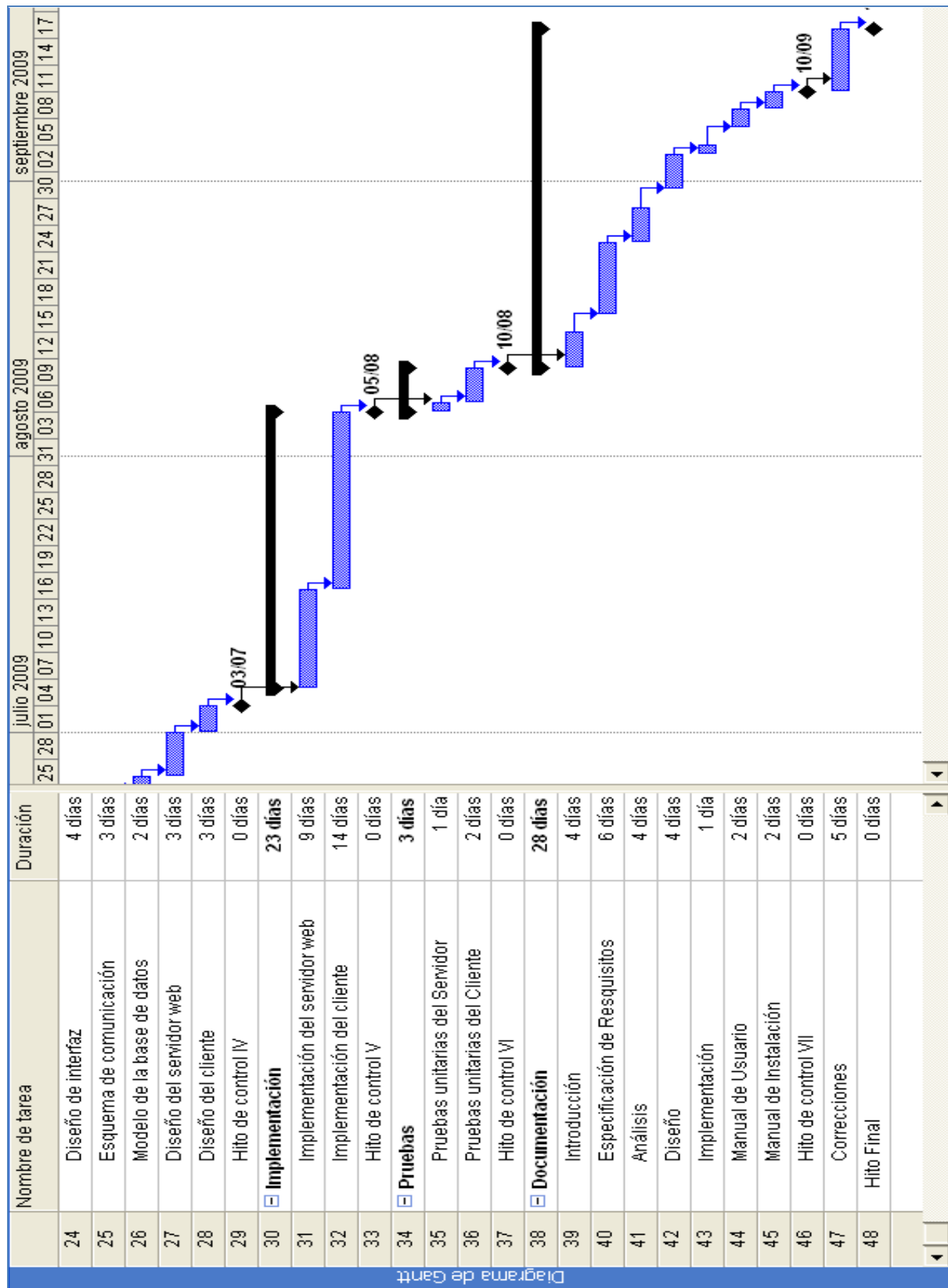


Figura 19b. Planificación del Proyecto (II)

Como resumen de las figuras 19a y 19b, se pueden extraer los siguientes cálculos de tiempo:

**Duración del proyecto en horas:**    *436 horas*, aproximadamente.

**Duración del proyecto:**                *109 días*.

**Horas aproximadas / día:**            *4 horas/día*, aproximadamente.

No se han contemplado el número de llamadas telefónicas realizadas ni el número de emails intercambiados con la tutora, Fuensanta Medina Domínguez.

### 3 ANÁLISIS

Esta sección muestra el comportamiento de la herramienta, observado como una caja negra, de manera visual mediante el Modelo Conceptual. Se trata de un Diagrama de Estructura Estática que consiste en identificar y representar los conceptos reales, no de componentes software, que conforman el dominio del proyecto y su objetivo es aumentar la comprensión del problema

#### 3.1 MODELO CONCEPTUAL

En la siguiente representación del *Modelo Conceptual* se exponen los conceptos, observados del mundo real, que tienen relevancia dentro de los límites o dominio del presente proyecto.

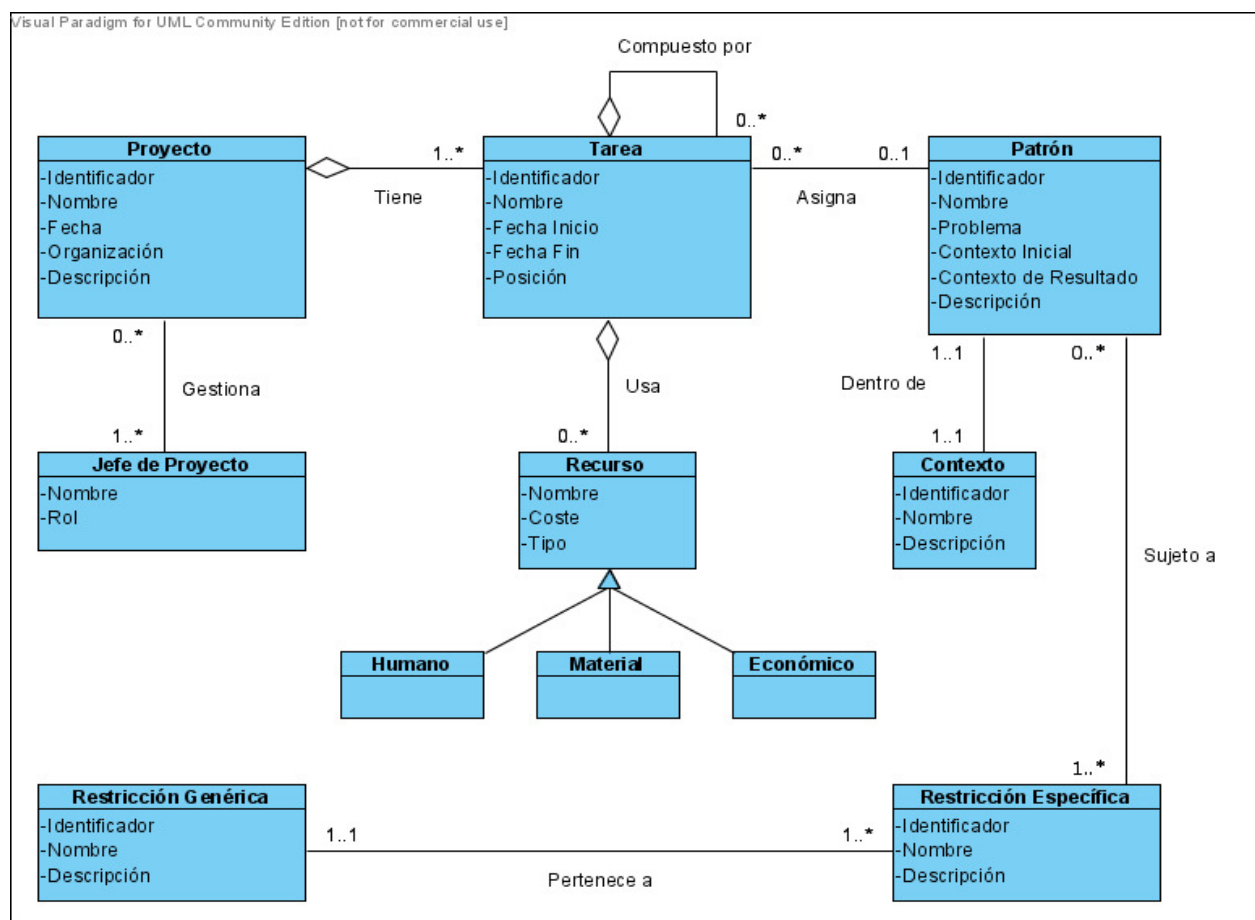


Figura 20. Modelo Conceptual



De la figura 29 se puede deducir que un Jefe de Proyecto puede gestionar varios proyectos y que estos están formados por tareas, a su vez compuestas por otras sub-tareas. Las tareas del proyecto pueden tener asociado un Patrón de Producto, el cual les aporta conocimiento y recomendaciones para la aplicación de las Buenas Prácticas. Estos patrones surgen como consecuencia de la selección de varias características: Restricción Genérica, Restricción Específica y Contexto. De esta manera, se establece una agrupación de patrones en función de un ámbito determinado.

## 4 DISEÑO

Esta sección comprende el *Modelo o Diseño de la Herramienta Software* enfocado a la necesidad que se pretende aportar, aplicando algunas de las normas básicas de *Usabilidad* [Dix-HCI, 2003], y teniendo en cuenta las tecnologías de las que se dispusieron para alcanzar el éxito del proyecto.

A continuación, se listan los componentes esenciales del proyecto, diferenciados y caracterizados según la función que desarrollan dentro del mismo, así como una referencia detallada sobre el entorno donde se desenvuelven:

- *Entorno del Diseño*: contiene la definición de la interfaz de usuario que interactúa con el servidor de servicios. (Apartado 4.3)
- *Cliente de los Servicios*: contiene la definición de la interfaz de usuario que interactúa con el servidor de servicios. (Apartado 4.4)
- *Servidor de los Servicios Web (o Web Service)*: contiene todos los servicios públicos que pueden ser llamados de manera externa gracias al fichero WSDL. (Apartado 4.5)
- *Bases de Datos*: contiene la información relevante para el éxito del proyecto y que es manejada desde el Servidor de Servicios o Web Service. (Apartado 4.7)

Se emplea el diseño por capas (presentación, lógica de negocio y datos) con el objetivo de dividir la funcionalidad de la aplicación, aportando una alta cohesión, un bajo acoplamiento y ofreciendo facilidad a la hora de evolucionar la solución propuesta.

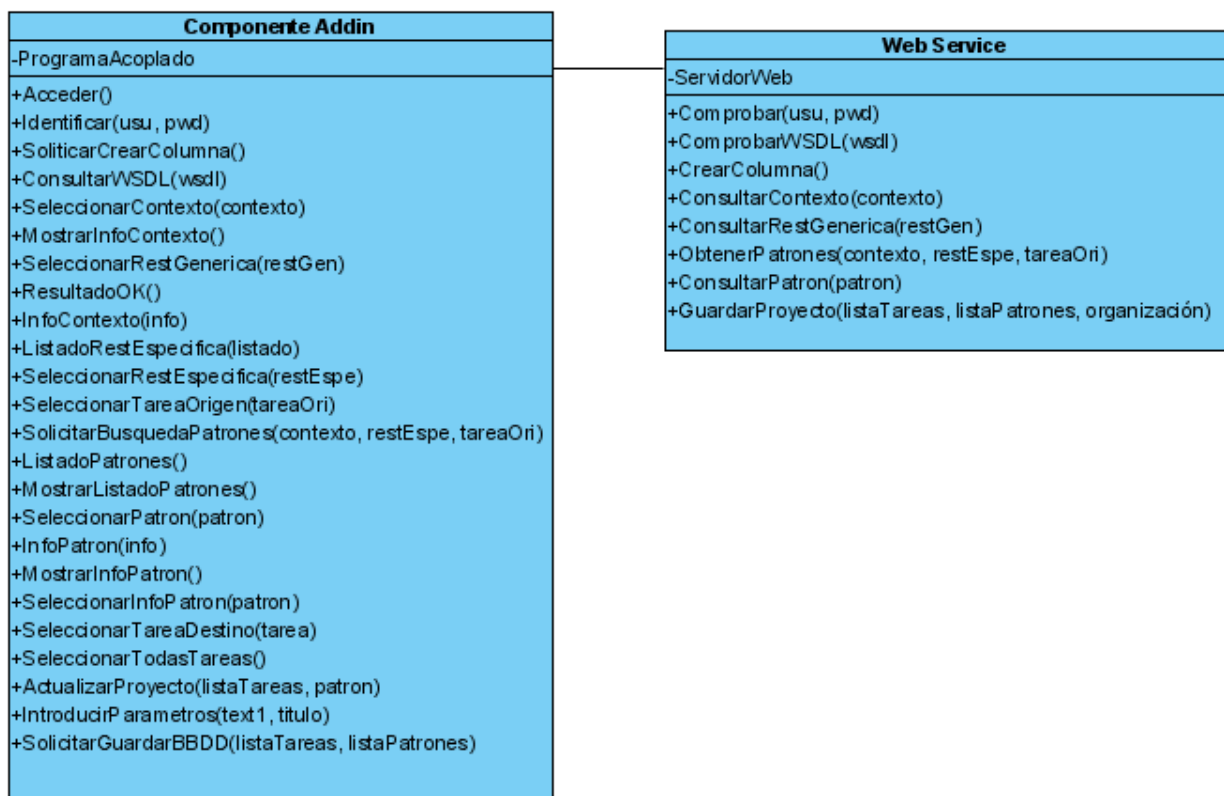
- Capa de presentación: aplicada al cliente software, representado por el componente Addin de MS-Project.
- Capa de negocio y Capa de datos: aplicadas a la implementación del Web Service.

El Cliente de los servicios necesita tener instalado el programa MSProject de Office, conexión a internet (o intranet, según donde se encuentre el Servicio Web) y un navegador web compatible con XML para realizar las peticiones al Web Service. No se requiere de un equipo especialmente potente.

Tanto el Web Service, como la base de datos y el servidor de aplicaciones web “GlassFish”, requieren instalarse y configurarse sobre un sistema operativo Linux Ubuntu v9.04.

## 4.1 DIAGRAMA DE CLASES DE DISEÑO

En el siguiente *Diagrama de Clases* se captura, especifica y muestra el vocabulario del proyecto (elementos, relaciones y estructura), de una manera clara y sencilla, fundamento del comportamiento de la herramienta software.



**Figura 21. Diagrama de Clases**

De la figura 30 se desprende la existencia de 3 clases: Componente Addin, Web Service y Navegador. La relación existente entre el Componente Addin y el Web Service es esencial para el buen funcionamiento de la herramienta creada y su acoplamiento se realiza de manera adecuada para llevar a buen término las funcionalidades definidas.

## 4.2 DIAGRAMAS DE SECUENCIA

Los diagramas de secuencia muestran los objetos como líneas de vida, así como sus interacciones en el tiempo, representadas por mensajes dibujados como flechas desde la línea de vida origen hasta la línea de vida destino y son buenos para mostrar la comunicación entre objetos por medio de mensajes entre ellos, pero no están pensados para mostrar lógicas de procedimientos complejos.

Nota: Se han descrito los “*cursos típicos de eventos*” de todos los Casos de Uso, ignorando sus posibles “cursos alternativos”.

### 4.2.1 Diagrama de Secuencia: CU Autenticación

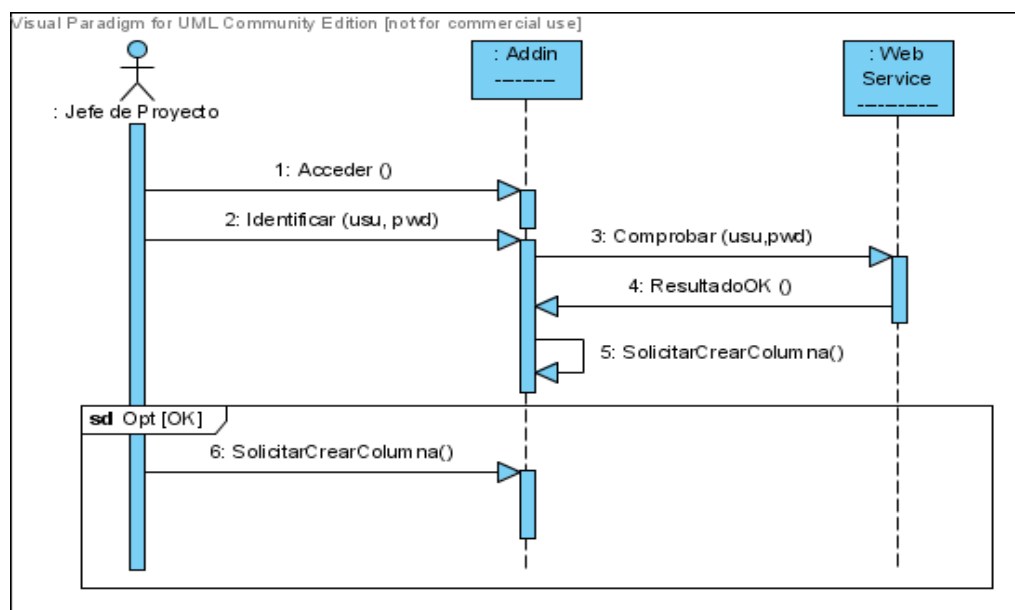
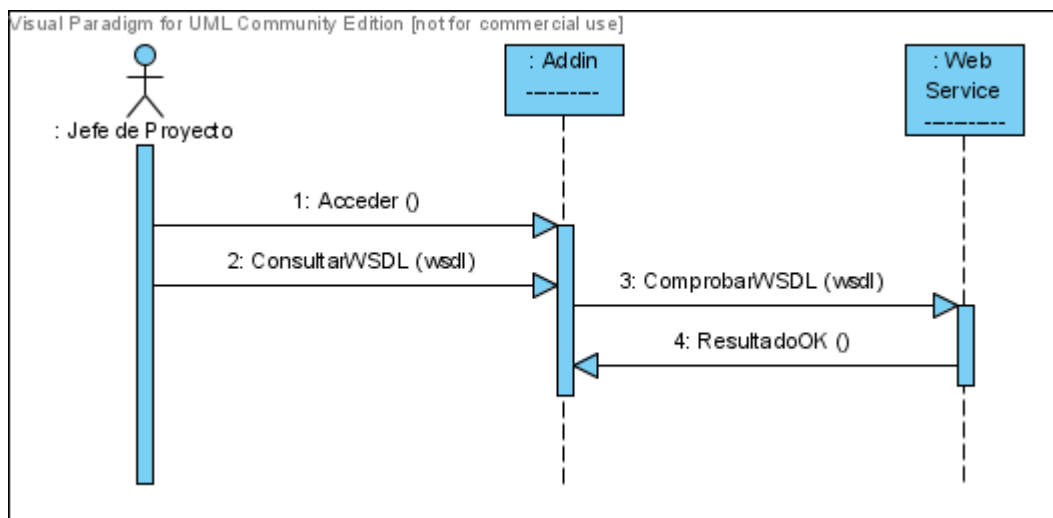


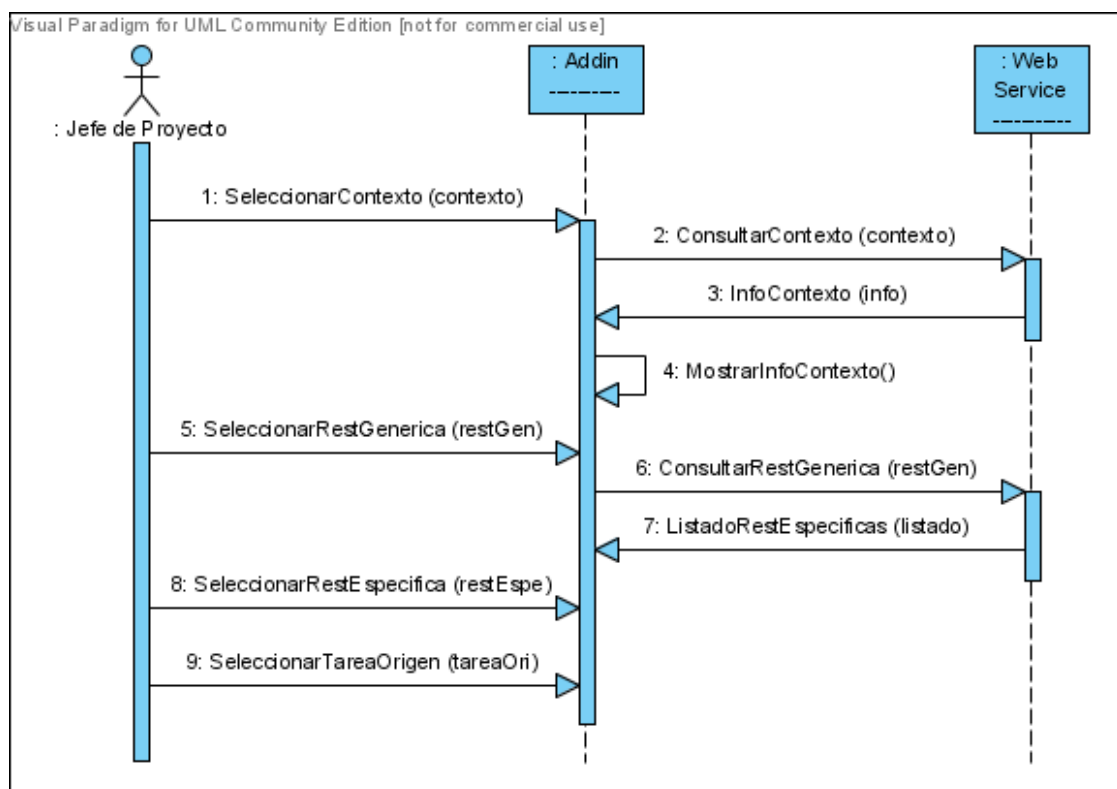
Figura 22. Diagrama de Secuencia: Identificación

#### 4.2.2 Diagrama de Secuencia: CU Probar Descriptor WSDL



**Figura 23. Diagrama de Secuencia: Probar Descriptor WSDL**

#### 4.2.3 Diagrama de Secuencia: CU Seleccionar Parámetros



**Figura 24. Diagrama de Secuencia: Seleccionar Restricciones**

#### 4.2.4 Diagrama de Secuencia: CU Realizar Búsqueda de Patrones

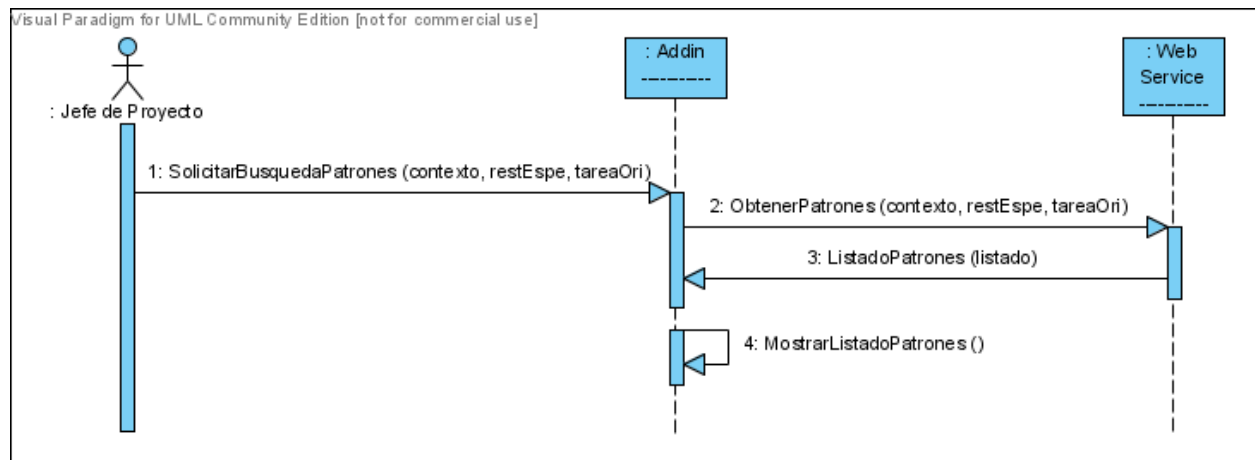


Figura 25. Diagrama de Secuencia: Realizar Búsqueda de Patrones

#### 4.2.5 Diagrama de Secuencia: CU Seleccionar Patrón

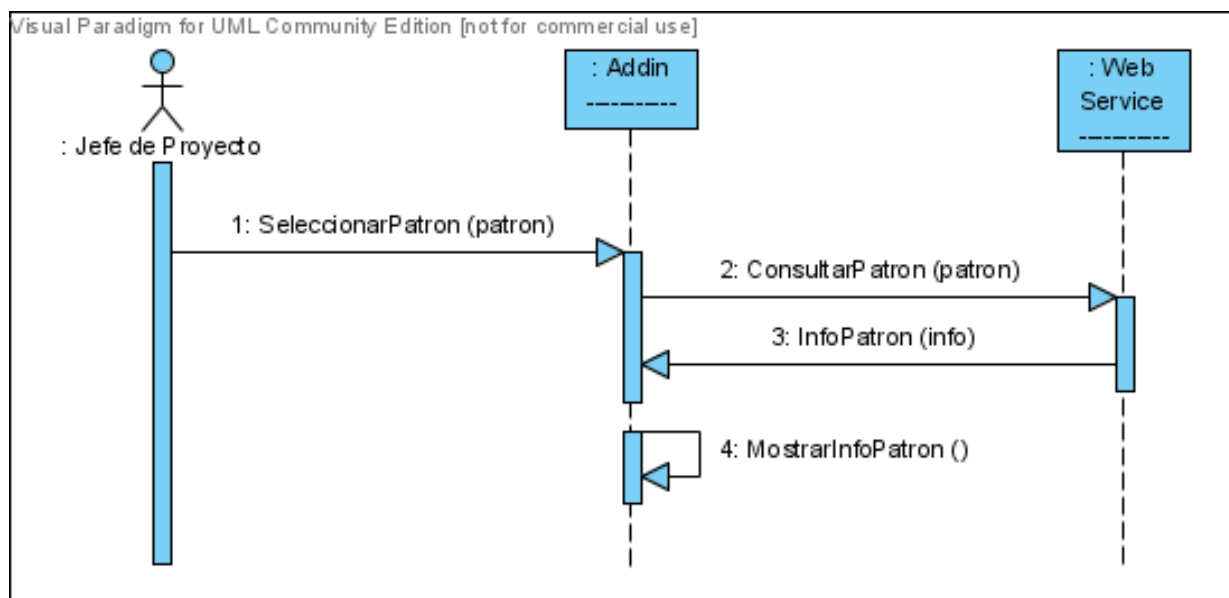
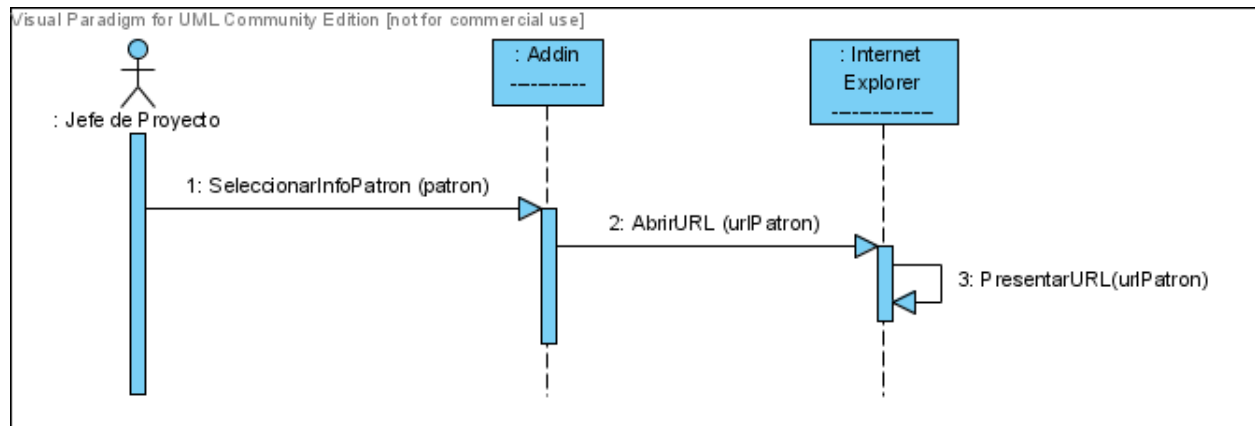


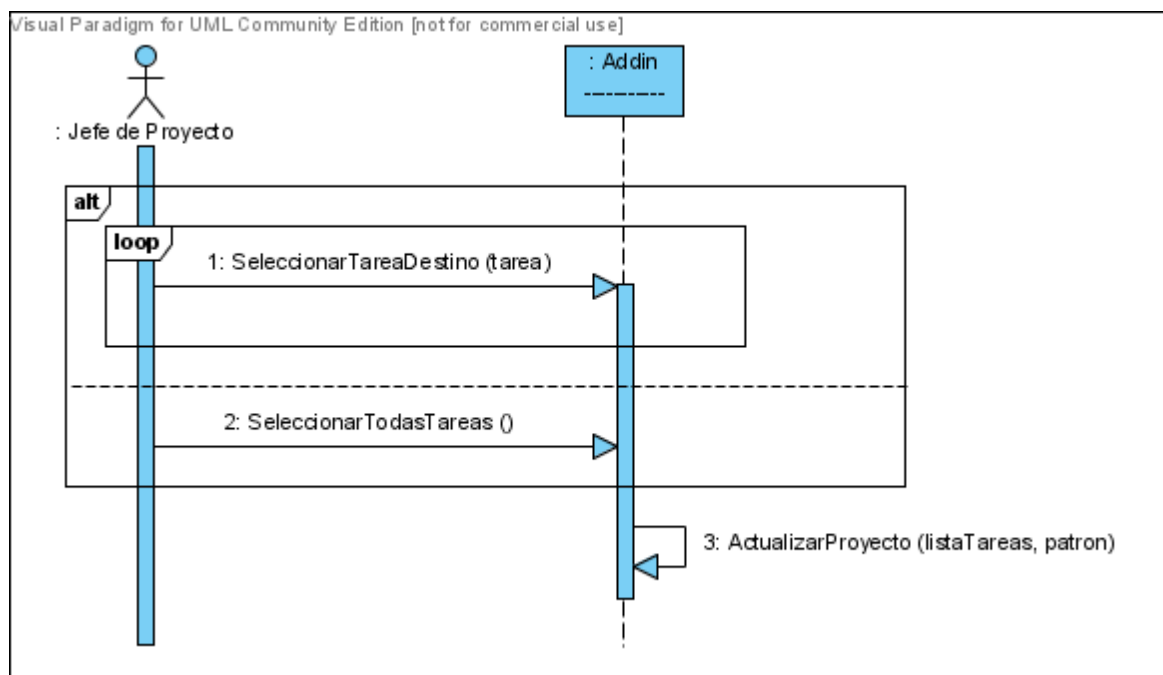
Figura 26. Diagrama de Secuencia: Seleccionar Patrón

#### 4.2.6 Diagrama de Secuencia: CU Consultar Información del Patrón



**Figura 27. Diagrama de Secuencia: Consultar Información del Patrón**

#### 4.2.7 Diagrama de Secuencia: CU Seleccionar Tareas Destino



**Figura 28. Diagrama de Secuencia: Seleccionar Tareas Destino**

#### 4.2.8 Diagrama de Secuencia: CU Crear Columna Destino

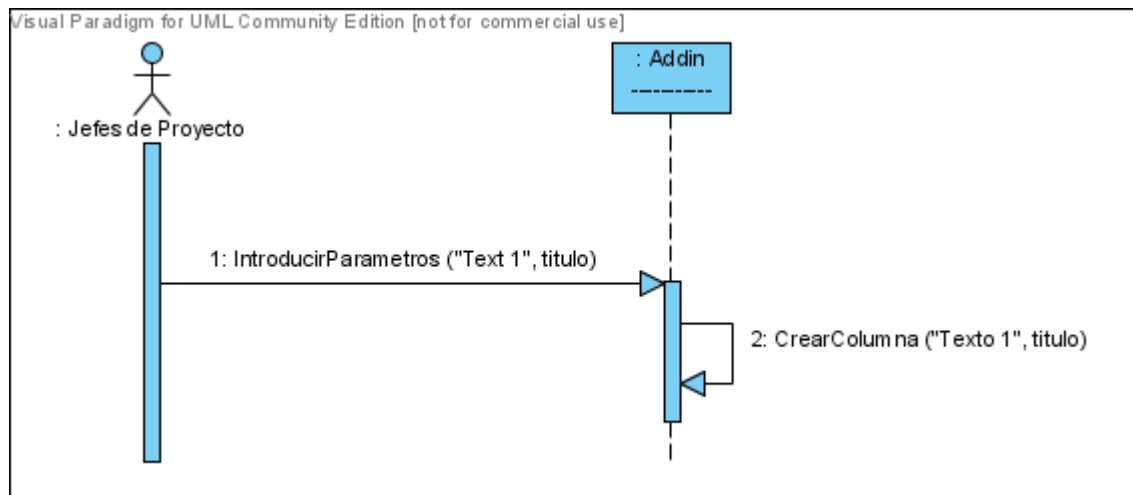


Figura 29. Diagrama de Secuencia: Crear Columna Destino

#### 4.2.9 Diagrama de Secuencia: CU Guardar proyecto en BBDD

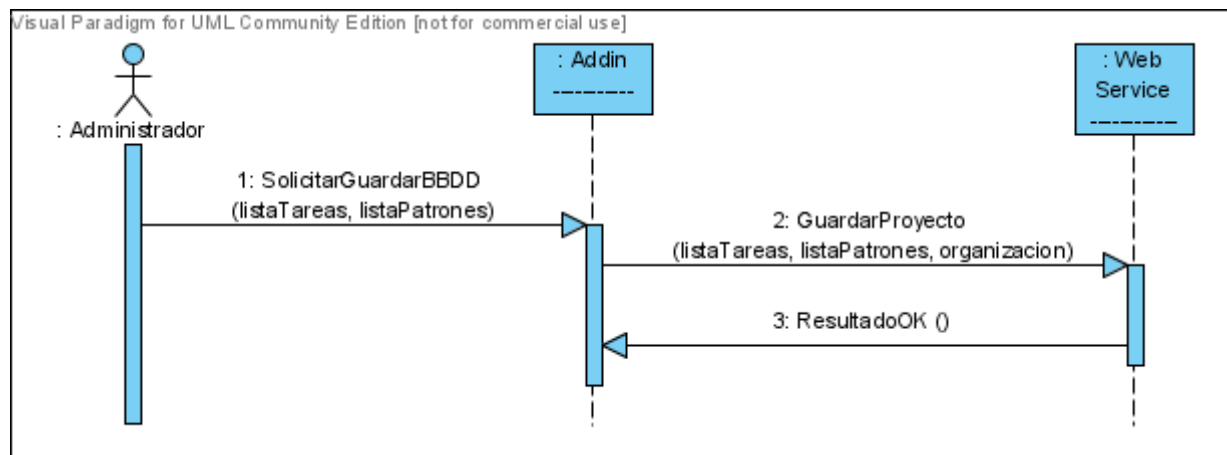


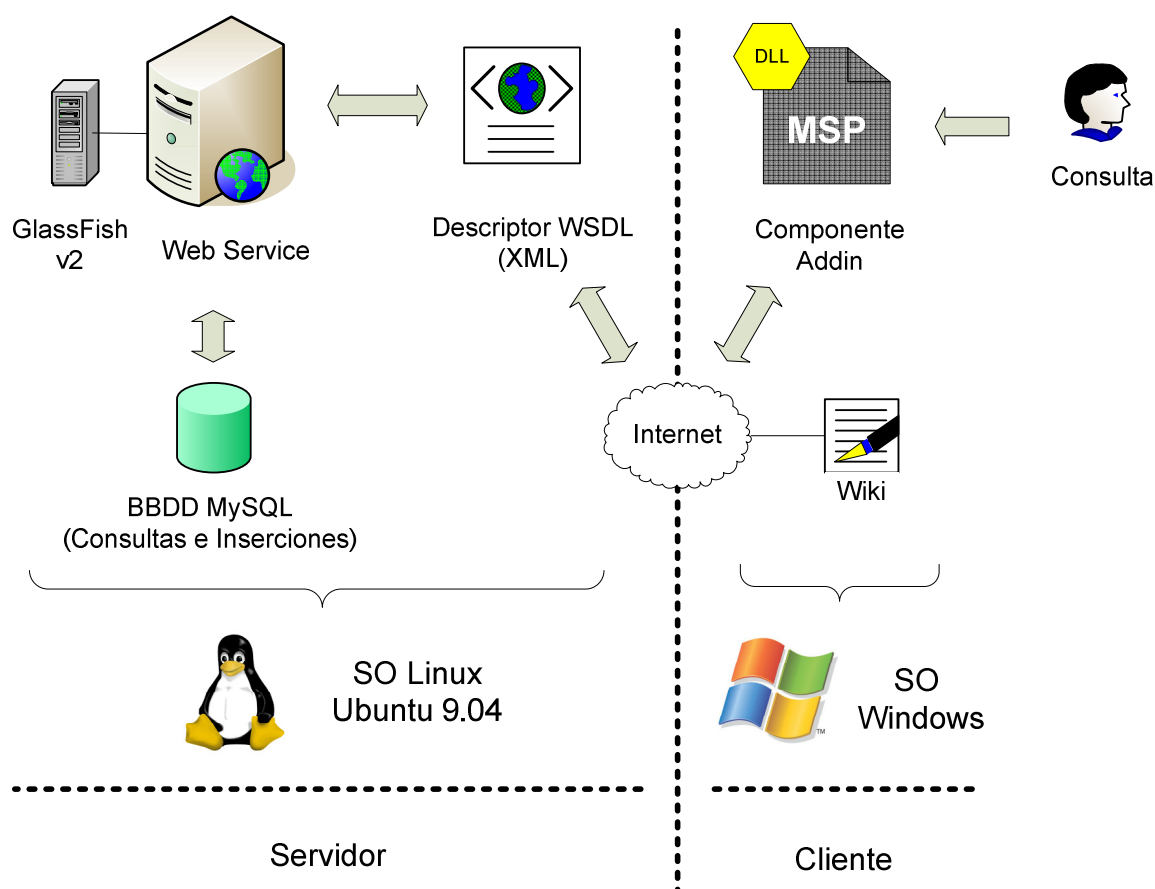
Figura 30. Diagrama de Secuencia: Guardar proyecto en BBDD



### 4.3 ENTORNO DEL DISEÑO – ESQUEMA DE COMUNICACIÓN

La herramienta se ha diseñado de tal manera que todos los elementos se puedan comunicar de manera orquestada y acorde a la funcionalidad establecida para ellos.

Visualizando la figura 31 podemos hacernos una idea más clara de la relación existente entre ellos, así como del rol individual que desempeñan, con respecto al conjunto del sistema.



**Figura 31. Diseño: Esquema de Comunicación**

La figura 31 representa al Web Service (o Servidor de Servicios Web) como un equipo Servidor Web, el cual se encuentra operativo gracias al Servidor de Aplicaciones Web “GlassFish”, en nuestro caso, versión 2. Este último, representado como un servidor de menor tamaño, enlazado al Web Service. Además, el Web Service se comunica directamente con el Gestor de Base de Datos MySQL, al que realiza peticiones de consulta de patrones e inserciones

de datos (tareas, patrones asociados, etc....), obtenidos de la planificación registrada sobre el programa MS-Project.

Todos estos elementos se hallan instalados sobre el sistema operativo Linux Ubuntu, versión 9.04, un sistema operativo que destaca por poseer un buen rendimiento, ser práctico, sencillo, y ser gratuito.

Por el otro lado, nos encontramos la parte del cliente donde está instalada la aplicación MS-Project junto con el componente Addin, creado para la ocasión, ambos sobre el sistema operativo Windows (Windows XP, Windows Vista o Windows 7). Este Addin es el encargado de realizar las peticiones de consulta y almacenamiento al Web Service gracias al Descriptor de Servicios WSDL, fichero que contiene el listado y descripción de los servicios web permitidos para ser consumidos por los clientes y que está basado en el lenguaje XML, publicado con acceso desde Internet mediante el protocolo SOAP. [INET, 32]

#### **4.4 CLIENTE DE LOS SERVICIOS WEB**

Se ha procurado que la interfaz definida para el cliente, por medio del componente Addin, sea lo más clara y sencilla posible, manteniendo los principios básicos de usabilidad (feedback, mecanismos de escape, etc.), tratando de que sea efectivo, eficiente, seguro, útil, fácil de aprender y de recordar.

La interfaz del cliente se ha implementado con el lenguaje Visual Basic [Aitken, 2002], Lenguaje Orientado a Eventos, haciendo uso de diferentes formularios y módulos de propósito general y específico. A continuación se lista una serie de características tenidas en cuenta durante el proceso de diseño de la interfaz del cliente:

- Se han introducido algunos elementos puntuales de reacción o *feedback* que aportan información al Jefe de Proyecto sobre qué tipo de operación se está realizando y el estado del sistema.

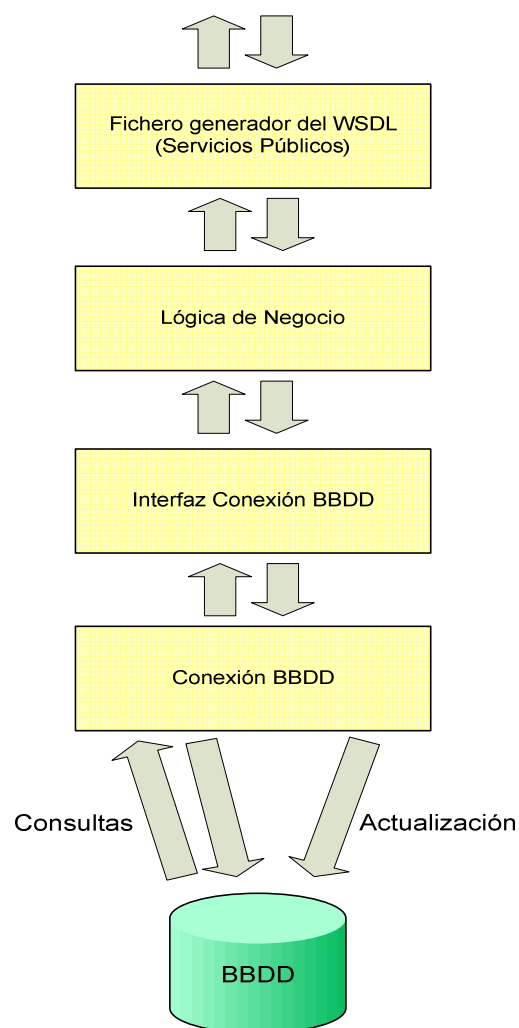
- Se ha permitido al Jefe de Proyecto el control del sistema, proporcionándole *mecanismos de escape* cuando se llegan a situaciones no deseadas, mediante mensajes visuales, en los que debe elegir entre las opciones disponibles, según el caso.
- Los *mensajes de error* están diseñados para ayudar a los Jefes de Proyecto a reconocer y diagnosticar la naturaleza del error, usando un lenguaje sencillo y apropiado para facilitar su comprensión.
- Se ha pretendido que el diseño fuera *minimalista*, eliminando aquella información que no fuera relevante.
- De igual manera, se ha pretendido que todos los controles fueran claramente *visibles*, con una buena representación sobre sus efectos.
- En todo momento se muestran las *restricciones* operativas existentes, las cuales dependen de la situación o momento en el que se encuentre el Jefe de Proyecto dentro de la herramienta.
- Se ha tratado de mantener la *consistencia y estandarización* lingüística para evitar que diferentes palabras tengan el mismo significado y esto pueda dar lugar a malas interpretaciones por parte de los Jefes de Proyecto, durante el empleo de la herramienta.
- Se busca *impedir cometer errores*, por parte de los Jefes de Proyecto, cuando sea posible durante el empleo de la herramienta.
- Se ha pretendido potenciar el *reconocimiento antes que el recuerdo* por medio de imágenes, posicionamiento de los elementos, así como la numeración de los pasos a seguir para alcanzar el objetivo deseado por el Jefe de Proyecto.

## 4.5 SERVIDOR DE LOS SERVICIOS WEB

El Servidor ofrece sus Servicios por medio de un fichero XML, con carácter público y ámbito en Internet, y no implementa ninguna interfaz gráfica para interactuar directamente con los usuarios.

Su interfaz viene delimitada por el nombre y los parámetros de entrada a los servicios y es el cliente quien gestiona los resultados devueltos por el servidor, dependiendo de dichos valores de entrada.

La figura 32 muestra el modelo de comunicación definido dentro del Web Service:



**Figura 32. Comunicación interna del Web Service**

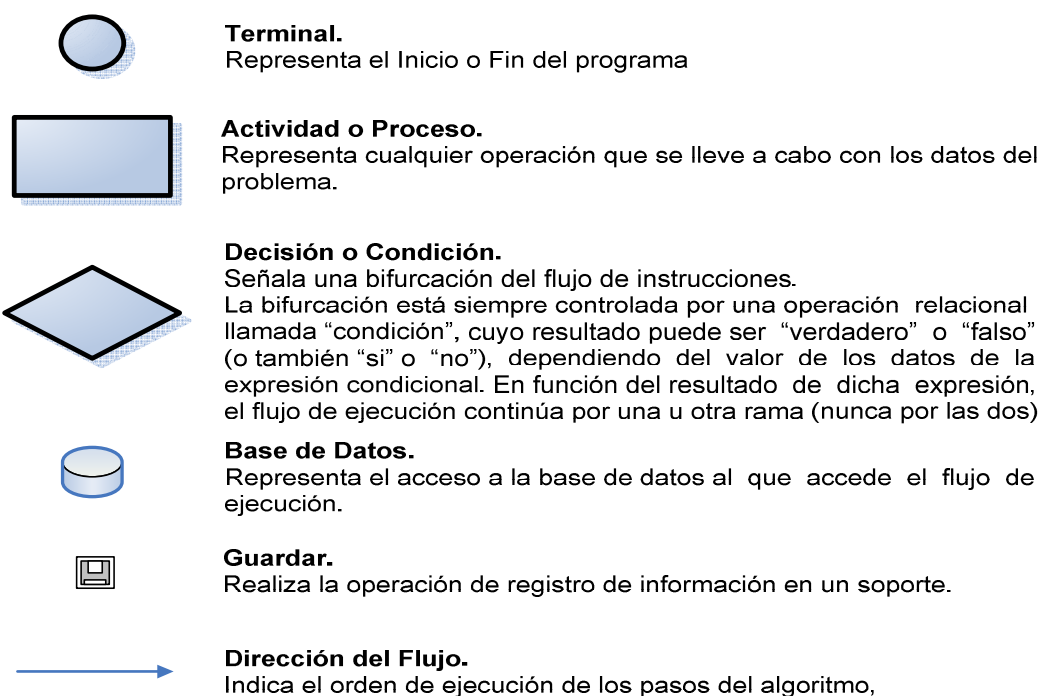
El fichero generador del WSDL es el encargado de hacer públicos y accesibles los servicios del Web Service, a través de Internet, así como de hacer llegar las peticiones desde el cliente al módulo o Capa de Lógica de Negocio. Estas peticiones, en su mayoría, llevan consigo peticiones a la base de datos, las cuales se realizan por medio del módulo o Capa de Datos.

Si no existió ninguna incidencia y la petición se gestionó correctamente, el módulo o Capa de Datos devuelve el resultado al módulo o Lógica de Negocio y éste, al fichero WSDL, a través del cual, se hace llegar la información solicitada por el mismo canal por el que se originó, vía Internet.

## 4.6 DIAGRAMA DE FLUJO

El “*Diagrama de Flujo*” se usa como representación gráfica de la secuencia de pasos que realiza la herramienta para llegar a obtener el listado de patrones adecuado a las tareas de la planificación, realizada por los Jefes de Proyecto, así como su vinculación. A continuación se muestra la leyenda para interpretar el Diagrama de Flujo:

### Leyenda para el Diagrama de Flujo



**Figura 33. Leyenda para el Diagrama de Flujo**

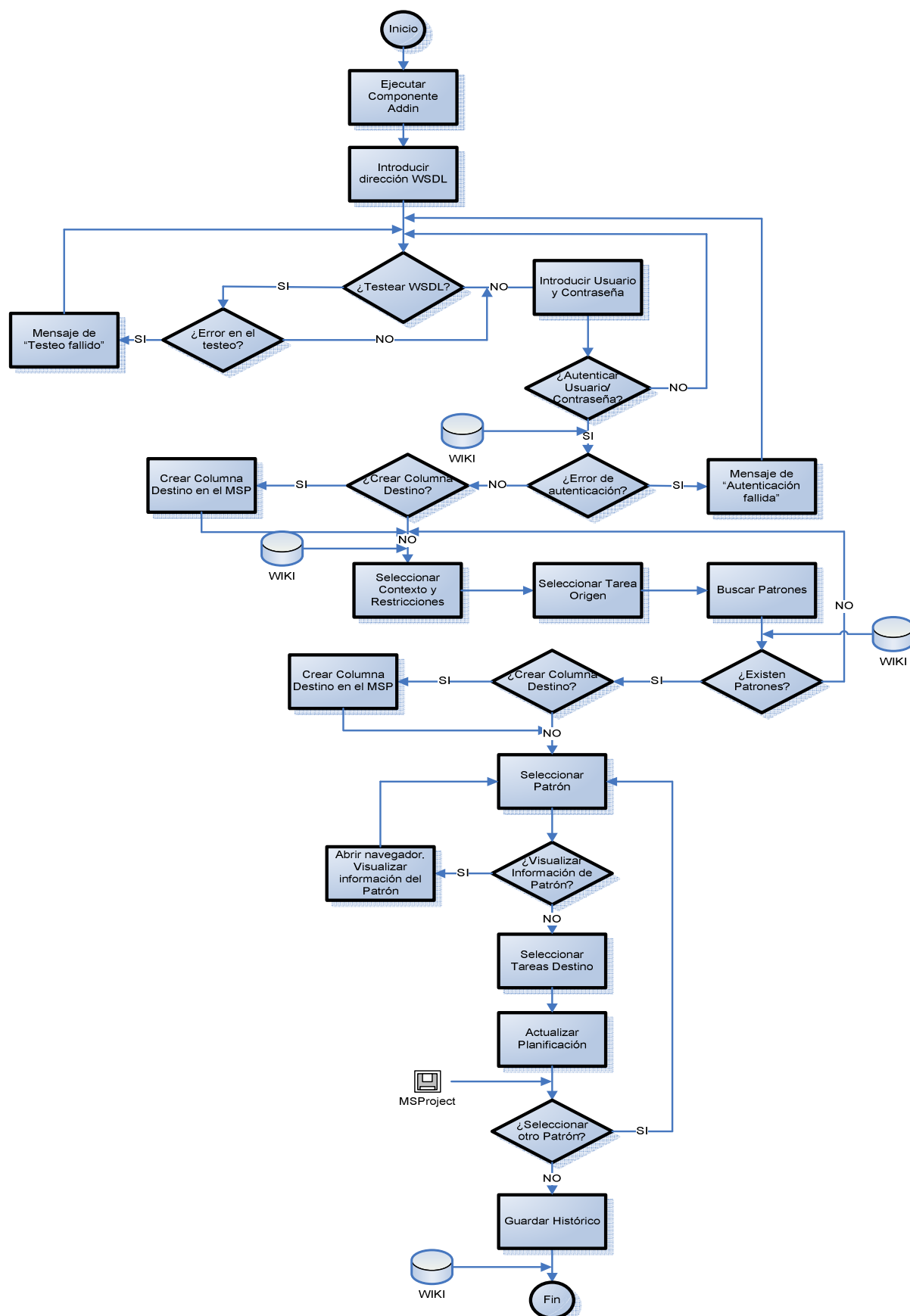
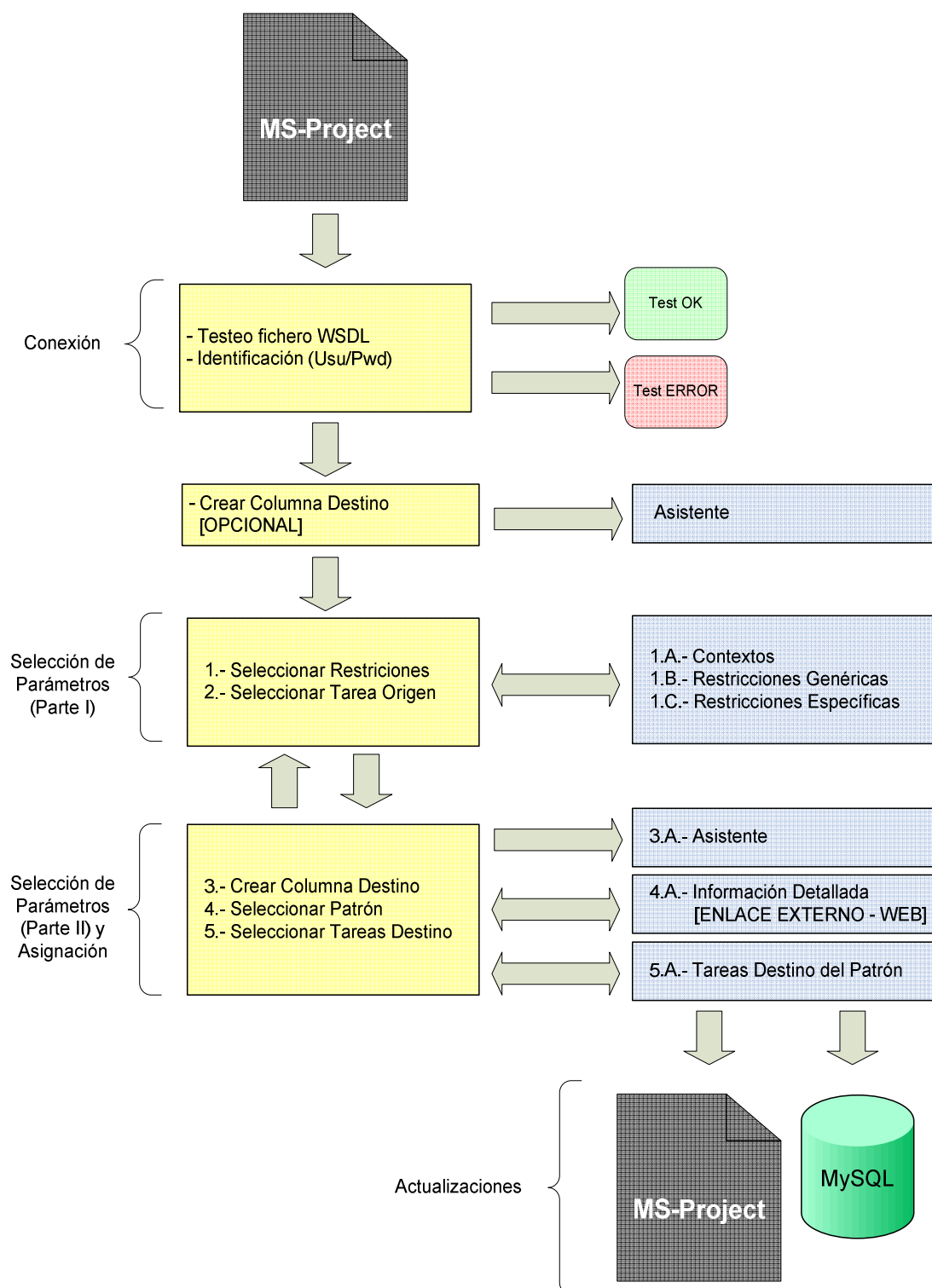


Figura 34. Diagrama de Flujo del funcionamiento de la aplicación

Representación del funcionamiento de la aplicación desde otra perspectiva para el cliente:



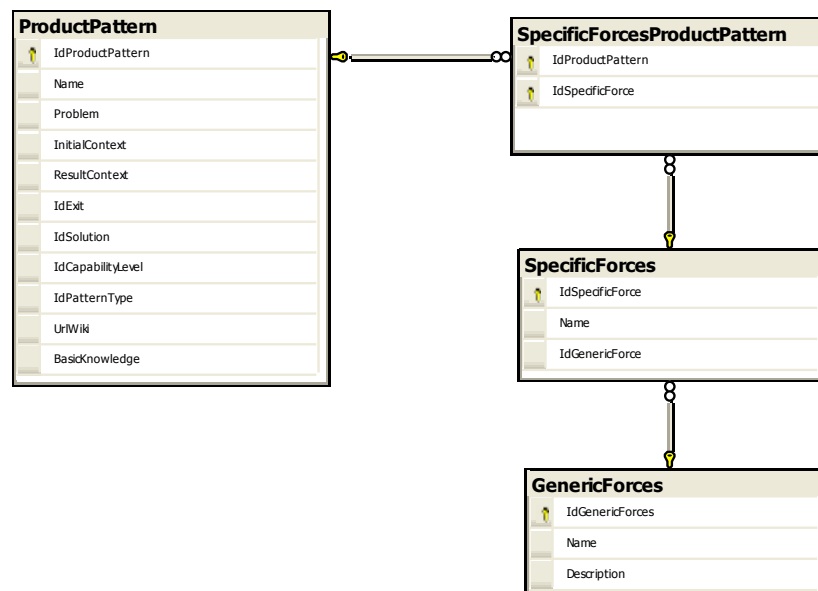
**Figura 35. Otra representación del funcionamiento de la aplicación**

## 4.7 MODELO DE DATOS

El modelo de datos está orientado a describir la Base de Datos, el cual tiene como objetivo almacenar y ofrecer toda la información relativa a los Patrones de Producto, conocimiento previamente analizado y estructurado, necesario para el uso de nuestra herramienta.

Las tablas esenciales de la herramienta software se centran en aquellas que contienen toda la información directa sobre los patrones de producto y sus restricciones específicas, éstas últimas obtenidas a partir de las restricciones genéricas. Otro aspecto a tener en cuenta son los atributos “*nombre*” y “*contexto*” de los patrones de producto, por medio de los cuales, y en combinación con la tarea seleccionada a la que se pretende asignar el patrón y el conocimiento asociado a este, se obtiene el listado de patrones relacionados más adecuado, así como su correspondiente dirección URL, almacenada en el campo “*URLWiki*”.

A continuación, se presenta el diagrama con las tablas de la base de datos que contienen toda la información referente a la búsqueda de patrones comentada en el párrafo precedente:

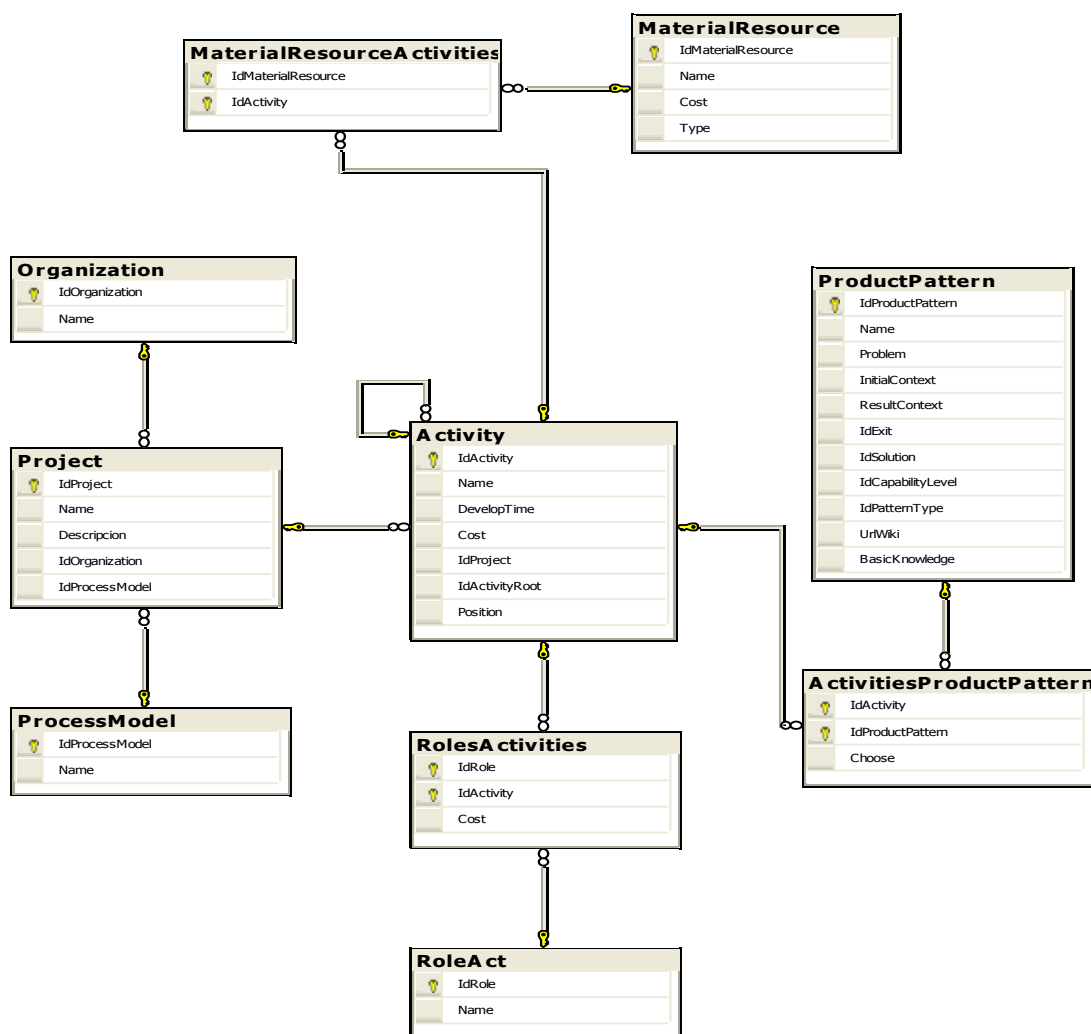


**Figura 36. Modelo de Datos: Tablas para la búsqueda de Patrones**



Por otra parte, se definieron varias tablas con el cometido de almacenar toda la información referente al proyecto en sí y a las asociaciones, creadas durante el uso de nuestra herramienta, entre las tareas y los patrones. Esta información tiene como objetivo servir de histórico y, en un futuro, se podrían implementar las funcionalidades necesarias para recuperar y manejar el detalle de las diferentes versiones de planificaciones guardadas. Otra posible futura funcionalidad de la herramienta podrá ser manejar dicha información para que sea usada como base de futuros estudios estadísticos.

El siguiente diagrama muestra las tablas usadas para almacenar la información del calendario o planificación generados por el programa MS-Project:



**Figura 37. Modelo de Datos: Tablas para guardar históricos de proyecto**

Diseño de la base de datos completa:

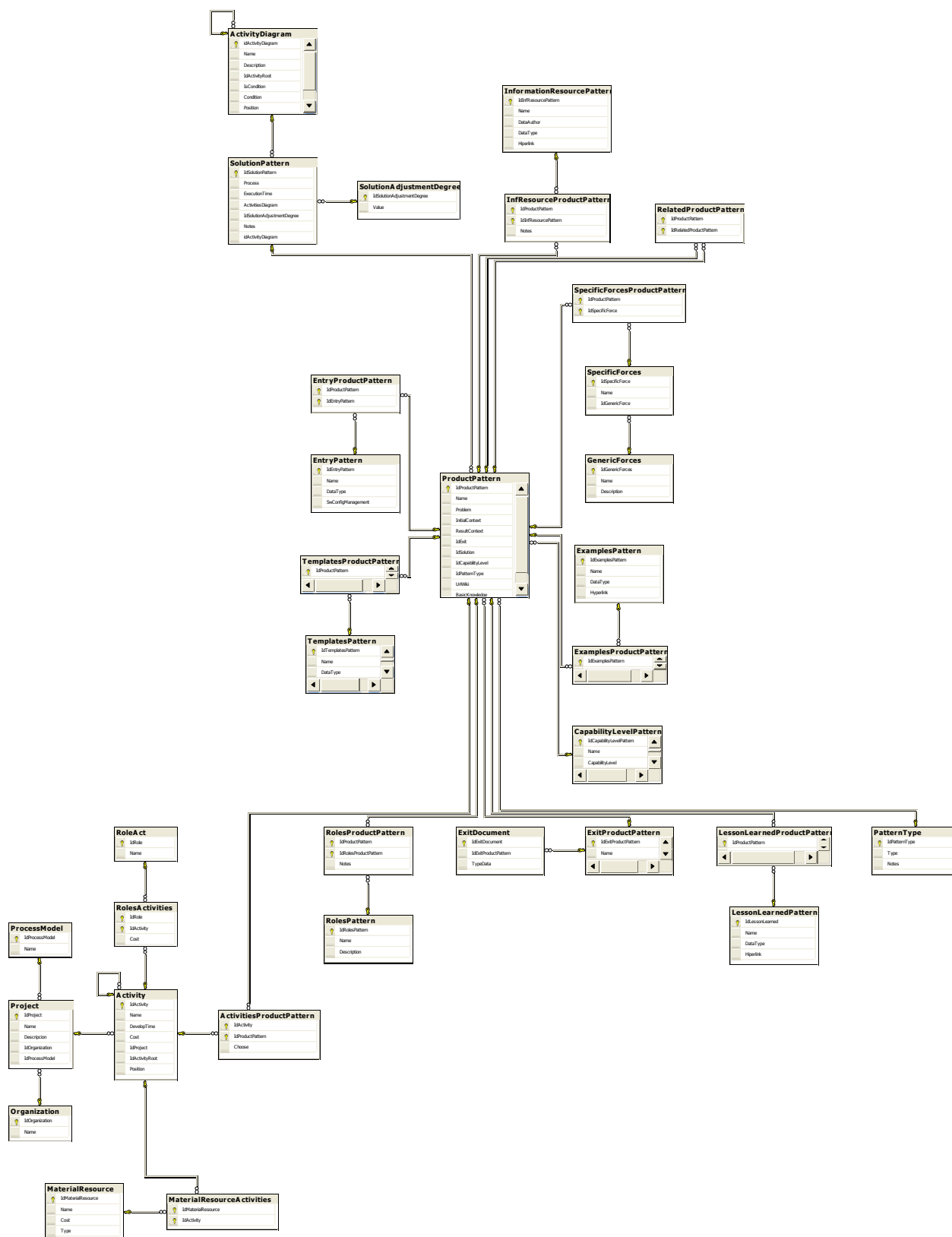


Figura 38. Modelo de Datos: Base de datos completa

## 5 IMPLEMENTACIÓN

En este apartado, se muestra la implementación en código fuente de uno de los casos de uso definidos en este documento. Dependiendo del componente (cliente o servidor) se usa un lenguaje de programación u otro (Visual Basic o Java, respectivamente), identificado en la cabecera de los métodos que se detallan a continuación:

### Caso de Uso: "Realizar Búsqueda de Patrones"

**Código del Cliente o Componente Addin** (Lenguaje de Programación: Visual Basic 6.0)

**Formulario:** frmSelParametros

**Evento:** Buscar\_Click

**Funcionalidad:** Realiza la llamada al formulario encargado de efectuar la petición de búsqueda de los Patrones de Producto por medio del servicio web público.

```
Private Sub cmdBuscar_Click()  
    On Error GoTo ErrManager  
  
    'Entorno gráfico del ratón y llamada al formulario encargado de realizar la búsqueda  
    Me.MousePointer = vbHourglass  
    frmBuscando.Show vbModal  
    Me.MousePointer = vbDefault  
  
    'Si la búsqueda obtuvo resultados, muestro el formulario para la selección del Patrones  
    If (UBound(gblArrUrlWiki) > 0) Then  
        Unload Me  
        frmPatrones.Show vbModal  
    Else  
        MsgBox "No patterns were found matching the selected search parameters", vbInformation  
        + vbOKOnly, "Search Result"  
    End If  
  
    Exit Sub  
  
ErrManager:  
    AddInErr Err 'Gestor de errores  
End Sub
```

**Código** del Cliente o Componente Addin (Lenguaje de Programación: Visual Basic 6.0)

**Formulario:** frmBuscando

**Evento:** Form\_Load

**Funcionalidad:** Realiza la llamada del servicio público del Web Service, destinado a la ejecución de la búsqueda de patrones de Producto, enviando los parámetros seleccionados previamente (usuario, contraseña, contexto, restricción específica y tarea origen).

```
Private Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)

Private Sub Form_Load()
    Dim result()      As String
    Dim tokens()      As String
    Dim numRegtrs, i, j As Integer
    Dim taskWords      As String

    On Error GoTo ErrManager

    ReDim tokens(1)
    PB.Value = PB.Value + 100
    i = 0
    j = 0

    'Obtengo las palabras de la tarea por separado (TOKENS)
    tokens = getTokens(frmSelParametros.lstActividades.Text)
    taskWords = Join(tokens, "-")

    PB.Value = PB.Value + 100

    'Conservo los parámetros de búsqueda (Contexto y Restricciones)
    gblContexto = frmSelParametros.lblContexto.Caption
    gblRestGenerica = frmSelParametros.lblRestGen.Caption
    gblRestEspecif = frmSelParametros.lblRestEsp.Caption

    'Llamada al servicio "getContext" para obtener el listado de CONTEXTOS
    result = gblSOAPClient.getPatterns(gblUsuario, gblContraseña, gblContexto, gblRestEspecif,
    taskWords)

    PB.Value = PB.Value + 100

    numRegtrs = UBound(result)

    ReDim gblArrResultContext(numRegtrs)
    ReDim gblArrUrlWiki(numRegtrs)

    PB.Value = PB.Value + 100
    frmPatrones.lstPatrones.Clear
```

```
'Se rellena la lista con los Patrones
For i = 0 To numRegtrs
    If (result(i) <> Empty) And (result(i) <> "0") Then
        frmPatrones.lstPatrones.AddItem result(i)
        i = i + 1
        frmPatrones.lstPatrones.ItemData(j) = result(i)
        i = i + 1
        gblArrResultContext(i - 2) = result(i)
        i = i + 1
        gblArrUrlWiki(i - 3) = result(i)
        j = j + 1
    End If

    PB.Value = PB.Value + 10
Next i

PB.Value = PB.Max

Exit Sub

ErrManager:
Select Case Err.Number
    Case 5415
        gblSubConectar (0)    'Error: perdida de conexión. Restaura la conexión
        Resume
    Case Else
        AddInErr Err          'Gestor de errores
End Select
End Sub
```

**Código** del Cliente o Componente Addin (Lenguaje de Programación: Visual Basic 6.0)

**Módulo:** ModuleSubs

**Procedimiento:** gblSubConectar

**Funcionalidad:** Configura los parámetros de conexión con el servidor de Servicios (Web Service) y establece dicha conexión. Permite realizar dos operaciones en función del parámetro de entrada: por un lado, verifica el acceso al fichero público WSDL; por otro lado, permite el acceso en la aplicación al usuario/contraseña introducidos.

```
Public Sub gblSubConectar(opcion As Integer)
    Dim result As Boolean

    On Error Resume Next

    'Creación del objeto SOAP para establecer la conexión
    Set gblSOAPClient = CreateObject("MSSOAP.SoapClient30")

    'Parámetros de configuración para el objeto SOAP recién creado
    gblSOAPClient.ClientProperty("ServerHTTPRequest") = True
    gblSOAPClient.ConnectorProperty("Timeout") = 60000
    gblSOAPClient.ConnectorProperty("UseSSL") = 0

    Err = 0    'Inicialización de la variable de control de conexión con el fichero WSDL

    'Llamada al servidor WSDL
    Call gblSOAPClient.mssoapinit(gblServidorWSDL)
    If Err <> 0 Then
        frmServidor.lblInfo.Caption = Err.Description
    End If

    'Selección de la opción de conexión: 0-> Prueba fichero WSDL; 1-> Conexión BBDD.
    If (opcion = 0) Then
        'Prueba de conexión con el servidor WSDL
        result = gblSOAPClient.testWSDL()
    ElseIf (opcion = 1) Then
        'Petición de confirmación del usuario/password introducidos
        result = gblSOAPClient.connectionRequest(gblUsuario, gblContraseña, 1)
    End If

    'Si la conexión fue exitosa, se actualiza la variable global que controlará el acceso de J.P.
    If (result = True) Then
        gblConectado = True
    Else
        gblConectado = False
    End If
End Sub
```

**Código** del Servidor de Servicios o Web Service (Lenguaje de Programación: Java)

**Clase:** WS\_Wiki

**Procedimiento:** getPatterns

**Funcionalidad:** Realiza la petición a la capa de lógica de negocio para obtener el listado de los Patrones de Producto que coinciden con los parámetros de búsqueda introducidos. Devuelve el resultado al cliente que inició la llamada del servicio.

```
package Pckg_Wiki;

import javax.jws.WebService;

@WebService
public class WS_Wiki {

    private LogicaNegocio ln = new LogicaNegocio();

    .....
    .....

    //Función que obtiene el listado de Patrones existentes en la base de datos seleccionada
    //Devuelve un array con los datos obtenidos
    public String [ ][ ] getPatterns (String usr, String pwd, String context, String
specificForce, String tokens) {
        String result[ ][ ] = null;
        String wordsTask[ ] = null;

        //Divide la cadena String en diferentes tokens separados por "-".Destino: Array
        wordsTask = tokens.split("-");

        //Llamada a la función que obtiene los Patrones según los parámetros de entrada
        result = ln.getPatterns(usr, pwd, context, specificForce, wordsTask);

        if (result == null) {
            result = new String[1][1];
            result[0][0] = Integer.toString(0);
        }

        return result;
    }

    .....
    .....

}
```

**Código** del Servidor de Servicios o Web Service (Lenguaje de Programación: Java)

**Clase:** LogicaNegocio

**Procedimiento:** getPatterns

**Funcionalidad:** Realiza la petición a la capa de datos para obtener el listado de Patrones de Producto que coinciden con los parámetros de búsqueda introducidos. Devuelve el resultado al método que inició la llamada.

```
package Pckg_Wiki;

import java.sql.ResultSet;
import java.sql.SQLException;

public class LogicaNegocio {

    private ConexionBDDDataSource con;
    private boolean conectado = false;

    .....
    .....
    //Función que solicita la conexión con la BBDD. Resultado: true/false
    public boolean connectionRequest (String usr, String pwd, int opcion) {
        con = new ConexionBDDDataSource();
        if (!con.conectarBD(usr, pwd, opcion)){
            conectado = false;
            return false;
        }
        else {
            conectado = true;
            return true;
        }
    }

    .....
    .....
    //Función que maneja los campos de los Patrones resultantes de la consulta.
    public String [ ][ ] getPatterns(String usr, String pwd, String context, String
specificForce, String wordsTask[ ]) {
        String result[ ][ ]= null;
        int i = 0, numRows = 0;

        conectado = connectionRequest(usr, pwd, 1);
```



```

if (conectado) {
    if (!con.existPatterns()) {
        return null;
    } else {
        try{
            ResultSet rs = con.getPatterns(context, specificForce, wordsTask);

            //Obtenemos el número de registros encontrados
            rs.last();
            numRows = rs.getRow();

            //Si existen coincidencias en la búsqueda, gestionamos el resultado
            if (numRows>0){
                result = new String[1][(numRows)*4];

                //Asignamos el valor de los registros obtenidos
                rs.first();
                result[0][i] = rs.getString("PPName");
                result[0][++i] = rs.getString("IdProductPattern");
                result[0][++i] = rs.getString("ResultContext");
                result[0][++i] = rs.getString("UrlWiki");
                rs.first();

                while (rs.next()) {
                    result[0][++i] = rs.getString("PPName");
                    result[0][++i] = rs.getString("IdProductPattern");
                    result[0][++i] = rs.getString("ResultContext");
                    result[0][++i] = rs.getString("UrlWiki");
                }

                con.finalize();
            }
        } catch (SQLException ex) {
            ex.printStackTrace();
            return null;
        } catch (Exception ex) {
            ex.printStackTrace();
            return null;
        }
        return result;
    }
} else
return null;
}

.....
.....
}

```

**Código** del Servidor de Servicios o Web Service (Lenguaje de Programación: Java)

**Clase:** ConexionBDDataSource

**Procedimiento:** getPatterns

**Funcionalidad:** Realiza la petición a la base de datos para obtener el listado de Patrones de Producto que coinciden con los parámetros de búsqueda introducidos. Devuelve el resultado al método de la capa de negocio que inició la llamada.

```
package Pckg_Wiki;

import java.sql.*;
import java.util.Calendar;

public class ConexionBDDataSource implements InterfaceConexionBD {

    private Connection con;
    private PreparedStatement ps;
    private ResultSet rs;

    .....

    .....
    //Función que realiza la conexión directa con la base de datos. Resultado: true/false
    public boolean conectarBD(String usr, String pwd, int opcion) {
        String url = null;

        if (opcion == 1) {
            url = "jdbc:mysql://localhost:3306/PIBOK_PB_ProductPatterns";
        } else if (opcion == 2){
            //url = "jdbc:mysql://localhost:3306/PIBOK_PB_ProjectBackup";
        }

        con = null;    //Inicialización de la variable de conexión

        try {
            // Se registra el Driver de MySQL
            Class.forName("com.mysql.jdbc.Driver").newInstance();

            // Se obtiene una conexión con la base de datos.
            con = DriverManager.getConnection(url, usr, pwd);
        } catch (Exception e) {
            e.printStackTrace();
            return false;
        }

        return true;
    }

    .....
}
```

```

//Función que realiza la consulta de patrones sobre la base de datos.
public ResultSet getPatterns(String context, String specificForce, String wordsTask[]) {
    int i; String sqlText = "";

    try {
        // Construcción de la Consulta SQL
        sqlText = " SELECT productpattern.IdProductPattern, ";
        sqlText = sqlText + " productpattern.Name as PPName, ";
        sqlText = sqlText + " productpattern.UrlWiki, ";
        sqlText = sqlText + " productpattern.ResultContext";
        sqlText = sqlText + " FROM productpattern INNER JOIN";
        sqlText = sqlText + " specificforcesproductpattern ON ";
        sqlText = sqlText + " productpattern.IdProductPattern = ";
        sqlText = sqlText + " specificforcesproductpattern.IdProductPattern INNER JOIN";
        sqlText = sqlText + " specificforces ON ";
        sqlText = sqlText + " specificforcesproductpattern.IdSpecificForce = ";
        sqlText = sqlText + " specificforces.IdSpecificForce";
        sqlText = sqlText + " WHERE (productpattern.InitialContext = " + context + ")";
        sqlText = sqlText + " AND (specificforces.Name = " + specificForce + ")";
        if (wordsTask.length > 0) {
            sqlText = sqlText + " AND (";
            for (i=0; i<wordsTask.length; i++){
                sqlText = sqlText + " (productpattern.Name LIKE '%" + wordsTask[i] + "%')";
                if ((i+1) < wordsTask.length) {
                    sqlText = sqlText + " OR";
                }
            }
            sqlText = sqlText + ")";
        }
        sqlText = sqlText + " ORDER BY PPName";

        ps = con.prepareStatement(sqlText);
        rs = ps.executeQuery(); //Ejecución de la consulta SQL

        if (rs.next())
        { return rs; }
    } catch (SQLException ex) {
        ex.printStackTrace();
        return null;
    } catch (Exception ex) {
        ex.printStackTrace();
        return null;
    }
    return null;
}
}

```

## 6 CONCLUSIONES Y TRABAJO FUTURO

### 6.1 CONCLUSIONES

El trabajo llevado a cabo durante el presente Proyecto Fin de Carrera ha supuesto tener que estudiar y aprender nuevas tecnologías, así como superar los diversos obstáculos que se fueron presentando, para alcanzar con éxito el objetivo deseado. Así pues:

- Se realizaron tantas sesiones de educación de requisitos con la tutora asignada como fueron necesarias, con el fin de alcanzar el suficiente conocimiento sobre el proyecto a desarrollar, lo cual sirvió para comprender y definir los objetivos perseguidos.
- Se aplicaron varios métodos y técnicas de la Ingeniería del Software para realizar un análisis, diseño, implementación y documentación apropiados de la herramienta en cuestión.
- Debido a las características del proyecto, se han tenido que estudiar nuevas tecnologías (Web Services, protocolo SOAP, MySQL, MSSQL SERVER) y lenguajes de programación (Java, Visual Basic, XML), diferentes a los impartidos en las asignaturas de la carrera universitaria. Esta experiencia ha sido muy útil para adquirir habilidades en la implementación de aplicaciones software, así como para potenciar la capacidad de adaptación ante nuevos retos.
- Se ha aplicado la programación por capas, implementando las capas de negocio y la capa de bases de datos en la parte del Servidor de Servicios (Web Service), y la capa de presentación o interfaz enteramente en el cliente de los servicios (o componente Addin). La principal ventaja del uso de la programación por capas es la implementación en varios niveles, lo que permite, en caso de que fuese necesario realizar algún cambio futuro, modificar solo el nivel requerido sin tener que revisar código entremezclado, con la dificultad de entendimiento que ello conlleva.

- Se analizaron las principales características de varias herramientas de desarrollo (IDE) existentes en el mercado (Kawa, Eclipse, NetBeans, JDeveloper) con el fin de poder identificar cual podría adaptarse mejor al propósito del presente trabajo fin de carrera. Finalmente, la selección quedo reducida a las últimas versiones de Eclipse [Daum, 2005] y de NetBeans, eligiendo la primera de ellas gracias a su facilidad de uso, de instalación, configuración y por tratarse de software libre.
- Se estudió la mejor manera de migrar la estructura e información de la base de datos existente en MSSQL SERVER al gestor de bases de datos MySQL haciendo uso, finalmente, de la herramienta “*MySQL GUI Tools v5*”. De este modo, se pasó de usar un gestor de base de datos bajo licencia a otro gestor de software libre, manteniendo durante todo el proceso la coherencia en la estructura de datos y en la información contenida en ella.
- Se buscó la manera de realizar la distribución, instalación y registro más sencilla e intuitiva posibles de la aplicación en los equipos cliente, realizándose la integración con MS-Project de manera transparente a los usuarios (por medio del uso de la herramienta “*Create Install v4.11.2*”). La elección de esta herramienta hizo que no fuera necesario invertir tiempo en su implementación, permitiendo centrar la atención en el desarrollo del resto de funcionalidades ofrecidas por el proyecto, y ofreciendo al usuario una visión amigable de la herramienta, durante todo el proceso instalación.
- La existencia de herramientas para generar instaladores de productos software redundan en ahorro de tiempo y esfuerzos en este tipo de proyectos, ya que permite dedicar más tiempo en la implementación de la herramienta software objetivo, sin perjuicio del objetivo perseguido por la herramienta. Así pues, se buscó la manera de realizar la distribución, instalación y registro más sencilla e intuitiva posibles de la aplicación en los equipos cliente, realizándose la integración con MS-Project de manera transparente a los usuarios, eligiendo el uso de la herramienta externa “*Create Install v4.11.2*”.

- Finalmente, se analizaron varios de los servidores de aplicaciones web más comunes del mercado, como son Apache y GlassFish, decidiéndonos por este último dada la fuerte relación existente entre dicho servidor de aplicaciones web y Java (ambos son propiedad de SUN Microsystems), lo que nos ofrecía una gran compatibilidad y acoplamiento entre ellos.

Así pues, el presente proyecto ha requerido de un forzoso y educativo periodo de tiempo para la adquisición de los nuevos conocimientos, tanto técnicos como conceptuales, imprescindibles para llevar a buen término los objetivos propuestos.

## **6.2 TRABAJO FUTURO**

Para futuras versiones se puede estudiar la viabilidad de funcionalidades que tengan como objetivo:

- Recuperar y manejar las diferentes versiones de planificaciones guardadas en la base de datos, almacenadas como históricos.
- La ampliación del número de características almacenadas en la base de datos (información sobre los plazos, recursos, costes, etc.), relativas a las planificaciones realizadas por los Jefes de Proyecto.
- La implementación de funcionalidades que manejen la información de los históricos almacenados en la base de datos, con el fin de que pueda ser usados como base a futuros estudios estadísticos.



## 7 BIBLIOGRAFÍA

### 7.1 MANUALES

- [Daum, 2005] “Professional Eclipse 3 for developers”  
Daum, Berthold. Ed. John Wiley & Sons (2005)  
(ISBN: 0470020059)
- [Recio-GO, 2007] Apuntes de la asignatura “Gestión de Proyectos”,  
Profesor: Miguel Recio.  
Curso 2007-2008.  
Universidad Carlos III de Madrid.
- [Gutiérrez, 2004] “MySQL”  
Gutiérrez Gallardo, Juan Diego. Ed. Anaya Multimedia (2004)  
(ISBN: 8441516839)
- [Dix-HCI, 2003] “Human-Computer Interaction”  
Alan Dix, Janet E. Finlay, Gregory D. Abowd, Russell and Russell  
Beale. Ed, Prentice Hall. 3rd Edition. (2003)  
(ISBN: 8428325456)
- [IEEE-830, 1998] “IEEE-STD-830-1998 : Especificaciones de los requisitos del software”
- [Sanchez-ISW2, 2007] Apuntes de la asignatura “Ingeniería del Software II”  
Profesores: María Isabel Sánchez Segura y Fuensanta Medina  
Domínguez.  
Curso 2007-2008.  
Universidad Carlos III de Madrid.
- [Chappell, 2002] “Java Web Services”  
Chappell, David (David A.). Ed. O'R'illy (2002)  
(ISBN: 0596002696)
- [Jacobson, 1992] “Object-Oriented Software Engineering: A Use Case Driven Approach”  
Jacobson I, Christenson M, Johnson P, Overgaard G.  
Addison-Wesley. Reading MA (1992)  
(ISBN: 0201544350)



- [Larman, 1999]            “UML y patrones: introducción al análisis y diseño orientado a objetos y al proceso unificado”  
C. Larman. Prentice Hall, 1999.  
(ISBN: 8420534382)
- [Chatfied, 2004]           “Microsoft Office Project 2003 paso a paso”  
Chatfied, Carl S.Ed. McGraw-Hill (2004)  
(ISBN: 8448140613)
- [Perez, 2004]            “MySQL para Windows y Linux”  
Pérez López, César. Ed. RA-MA (2004)  
(ISBN: 8478976019)
- [Pressman, 1998]        “Ingeniería del software. Un enfoque práctico”  
R. S. Pressman. 4ª Edición. McGrawHill (1998)  
(ISBN: 8448111869)
- [Piattini, 2003]           “Calidad en el desarrollo y mantenimiento del software”  
Piattini, M. G. – García, F.O. Ed. Ra-Ma. (2003)  
(ISBN: 8478975446)
- [Sommerville, 2005]     “Ingeniería de Software”  
Ian Sommerville. Ed. Pearson Educación. (2005)  
(ISBN: 8478290745)
- [Ferré, UML]            “Desarrollo Orientado a Objetos con UML”  
Xavier Ferré Grau (Facultad de Informática – UPM)  
María Isabel Sánchez Segura (Escuela Politécnica Superior – UC3M)
- [Aitken, 2002]           “Visual Basic 6 : manual completo de programación”  
Aitken, Peter G. Ed. Paraninfo (2002)  
(ISBN: 0130461091)<

## 7.2 ENLACES WEB

- [INET, 01] *Diagrama de Gantt.*  
<http://www.free-logistics.com/index.php/es/Fichas-Tecnicas/Gestion-de-Proyectos/Diagrama-de-Gantt.html>
- [INET, 02] *Método Pert.*  
[http://dis.um.es/~lopezquesada/documentos/FIS\\_0405/Teama8Pert.doc](http://dis.um.es/~lopezquesada/documentos/FIS_0405/Teama8Pert.doc)
- [INET, 03] *Método Pert.*  
<http://es.kioskea.net/contents/projet/pert.php3>
- [INET, 04] *Técnicas de Planeación.*  
<http://www.mitecnologico.com/Main/TecnicasDePlaneacion>
- [INET, 05] *Técnicas de Gestión de Proyectos.*  
<http://www.uazuay.edu.ec/estudios/sistemas/operativa2/GestiondeProyectos.ppt>
- [INET, 06] *Diagrama de Ishikawa.*  
<http://ictnet.es/system/files/Diagrama+de+Ishikawa.pdf>
- [INET, 07] *Wikipedia: Diagrama de Ishikawa.*  
[http://es.wikipedia.org/wiki/Diagrama\\_de\\_Ishikawa](http://es.wikipedia.org/wiki/Diagrama_de_Ishikawa)
- [INET, 08] *Wikipedia: Diagrama de Pareto.*  
[http://es.wikipedia.org/wiki/Diagrama\\_de\\_Pareto](http://es.wikipedia.org/wiki/Diagrama_de_Pareto)
- [INET, 09] *Fundamentos de la Ing. del Software. Tema 8: Método Roy.*  
[http://dis.um.es/~lopezquesada/documentos/FIS\\_0405/Tema8Roy.ppt](http://dis.um.es/~lopezquesada/documentos/FIS_0405/Tema8Roy.ppt)  
(Transp.2)
- [INET, 10] *Fundamentos de la Ing. del Software. Tema 8: Método Roy.*  
[http://dis.um.es/~lopezquesada/documentos/FIS\\_0405/Tema8Roy.ppt](http://dis.um.es/~lopezquesada/documentos/FIS_0405/Tema8Roy.ppt)  
(Transp.26)
- [INET, 11] *Herramientas CASE.*  
<http://www.monografias.com/trabajos73/herramientas-case-proceso-desarrollo-software/herramientas-case-proceso-desarrollo-software2.shtml>
- Wikipedia. Project Management Institute.*  
[http://es.wikipedia.org/wiki/Project\\_Management\\_Institute](http://es.wikipedia.org/wiki/Project_Management_Institute)
- Official Site: Project Management Institute.*  
<http://www.pmi.org/Pages/default.aspx>
- [INET, 12] *Wikipedia: Microsoft Project.*  
[http://es.wikipedia.org/wiki/Microsoft\\_Project](http://es.wikipedia.org/wiki/Microsoft_Project)

- 
- [INET, 13] *Wikipedia: Microsoft Visio.*  
[http://es.wikipedia.org/wiki/Microsoft\\_Visio](http://es.wikipedia.org/wiki/Microsoft_Visio)
- [INET, 14] *Wikipedia: Diagrama de Gantt.*  
[http://es.wikipedia.org/wiki/Diagrama\\_de\\_Gantt](http://es.wikipedia.org/wiki/Diagrama_de_Gantt)
- [INET, 15] *Herramientas Freeware para hacer Diagramas de Gantt.*  
<http://diego-seminariodeinvestigacion1.blogspot.com/2009/01/conceptos.html>
- [INET, 16] *Official Site: Knowledge Management Key (KMKey).*  
<http://www.kmkey.com/>
- [INET, 17] *Planificación de proyectos mediante GanttProject.*  
<http://www.aplicacionesempresariales.com/ganttproject-planifica-tus-propios-proyectos.html>
- [INET, 18] *Herramientas de uso libre para gestión de proyectos.*  
<http://www.navegapolis.net/content/view/56/49/>
- [INET, 19] *Gestión de proyectos: Planner.*  
[http://wiki.canaima.softwarelibre.gob.ve/wiki/index.php/Gesti%C3%B3n\\_de\\_proyectos\\_Planner](http://wiki.canaima.softwarelibre.gob.ve/wiki/index.php/Gesti%C3%B3n_de_proyectos_Planner)
- [INET, 20] *Official Site: TaskJuggler.*  
<http://www.taskjuggler.org/>
- [INET, 21] *Gestión de proyectos: Dia.*  
<http://www.aplicacionesempresariales.com/dia-un-editor-de-diagramas-gratuito-y-compatible-con-visio.html>
- [INET, 22] *Open Workbench, Software de Proyectos.*  
<http://www.aplicacionesempresariales.com/open-workbench-software-de-proyectos.html>
- [INET, 23] *Gestión de Proyectos.*  
<http://www.gestiopolis.com/recursos/documentos/fulldocs/ger/gestioproyecto.htm>
- [INET, 25] *Wikipedia: Dynamic Linking Library.*  
[http://es.wikipedia.org/wiki/Dynamic\\_Linking\\_Library](http://es.wikipedia.org/wiki/Dynamic_Linking_Library)
- [INET, 26] *Soporte Microsoft: Definición y explicación de un archivo .DLL.*  
<http://support.microsoft.com/kb/87934/es>
- [INET, 27] *Introducción a los Web Services.*  
<http://150.185.75.30/atiwiki/index.php/WEBSERVICES>

- [INET, 28] *Definición de WSDL.*  
<http://www.alegsa.com.ar/Dic/wsdl.php>
- [INET, 29] *Wikipedia: Definición de Técnica.*  
<http://es.wikipedia.org/wiki/T%C3%A9cnica>
- [INET, 30] *Wikipedia: Project Management Body of Knowledge.*  
[http://es.wikipedia.org/wiki/Project\\_Management\\_Body\\_of\\_Knowledge](http://es.wikipedia.org/wiki/Project_Management_Body_of_Knowledge)
- [INET, 31] *Servidor de Aplicaciones Glassfish (Java EE).*  
<http://michael-gutierrez.blogspot.com/2009/08/servidor-de-aplicaciones-java-ee-en.html>
- [INET, 32] *Wikipedia: SOAP*  
<http://es.wikipedia.org/wiki/SOAP>
- [INET, 33] *Glosario de Términos (varios.)*  
<http://e-amanecer.com/licenciatura/glosario.html>
- [INET, 34] *Fundamentos: Calidad del Software.*  
<http://www.slideshare.net/zamanthag/fundamentos-tecnologia-internet>
- [INET, 35] *Calidad en Ingeniería del Software.*  
[http://is.umb.edu.co/aulaumb/file.php/37/Calidad\\_de\\_software.PPT](http://is.umb.edu.co/aulaumb/file.php/37/Calidad_de_software.PPT)
- [INET, 36] *Wikipedia: Wiki.*  
<http://es.wikipedia.org/wiki/Wiki>



## **ANEXO A. MANUAL DE INSTALACIÓN**

---

## **1 INSTALACIÓN EN EL SERVIDOR**

### **1.1 REQUERIMIENTOS MÍNIMOS**

Antes de instalar la herramienta y poder usarla, debemos asegurarnos de contar con:

- Sistema Operativo: Linux Ubuntu v9.04
- Acceso a Internet

El equipo servidor debe disponer de, al menos, un microprocesador con velocidad de 2 Gigahercios (GHz) de procesamiento, una memoria RAM de 2 Gigabytes (GB) y un espacio en el disco duro de 500 Megabytes (MB).

### **1.2 INSTALACIÓN DEL HARDWARE**

No se requiere la instalación de ningún componente hardware específico para el uso de la herramienta.

### **1.3 INSTALACIÓN DEL SOFTWARE**

La instalación del software requiere varios componentes imprescindibles. Estos son los siguientes:

- Gestor de Bases de datos: MySQL
- Entorno de Desarrollo Integrado: Eclipse
- Servidor de Aplicaciones Web: GlassFish v2

### 1.3.1 Gestor de Base de Datos: MySQL

Se necesita disponer del programa Gestor de Bases de Datos “*MySQL*”, versión 5.1, el cual se puede obtener de manera gratuita, por medio de la página oficial de SUN Microsystems: <http://dev.mysql.com/downloads>.

Una vez lo tengamos en nuestro equipo, solo tenemos que iniciar su programa instalador y seguir los pasos de la instalación.

Para crear la estructura de la base de datos, así como para insertar la información necesaria dentro de ella, que será usada por la herramienta, se debe importar un fichero o Script con su contenido a través de un comando introducido en una ventana de consola abierta.

La estructura del comando, para la inserción de la estructura de la base de datos y de la información, es la siguiente:

```
mysql -u NOMBRE_USUARIO -p NOMBRE_BBDD_DESTINO < NOMBRE_SCRIPT.sql
```

Donde:

- u Nombre del usuario del sistema operativo, con permisos para realizar la operación
- p Nombre de la base de datos destino de la actualización del Script.
- < Nombre del fichero Script, que contiene toda la sintaxis SQL necesaria.

Posteriormente, se solicitará teclear la contraseña para el usuario introducido en la línea de comandos, con el fin de comprobar que disponga de los permisos necesarios para realizar la operación. En caso contrario, se informará que la operación no le está permitida a dicho usuario o que la contraseña no coincide.

Ejemplo:

```
C:\>mysql -u root -p PIBOK_PB_ProductPatterns < respaldo.sql
```

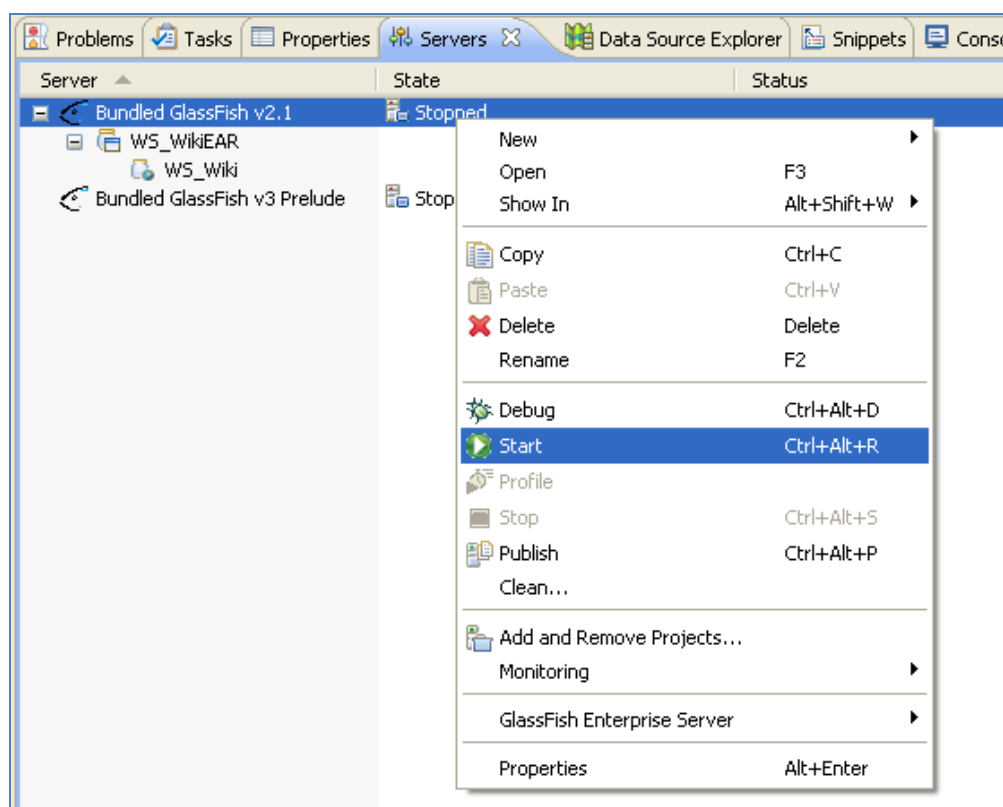


### 1.3.2 Entorno de Desarrollo Integrado: Eclipse

Se necesita disponer del Entorno de Desarrollo Integrado (IDE, Integrated Development Environment) “*GlassFish Tools Bundle For Eclipse*”, versión 1,0, el cual se puede obtener de manera gratuita, por medio de la página oficial de SUN Microsystems: <http://download.java.net/glassfish/eclipse>.

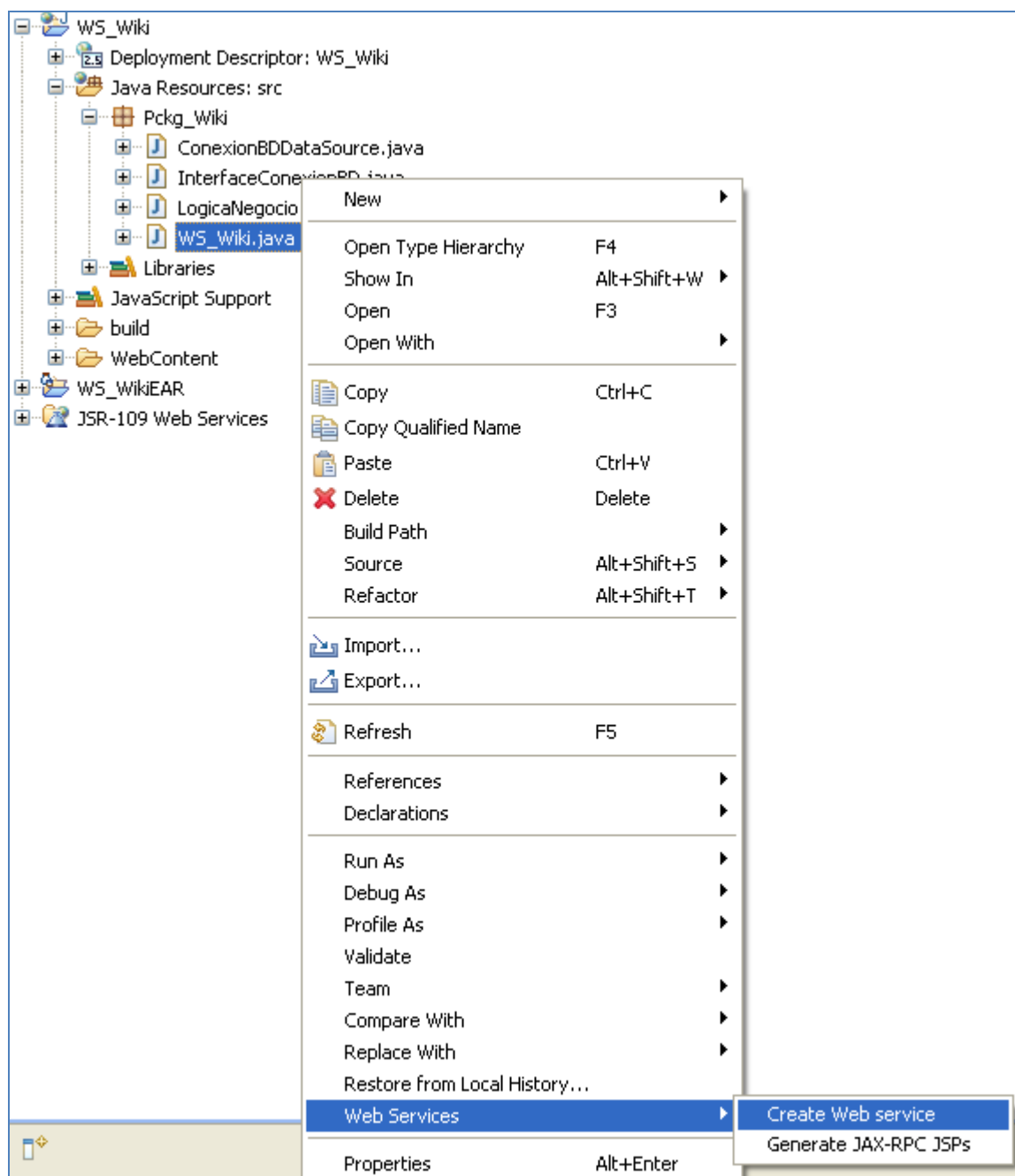
Una vez lo tengamos en nuestro equipo, solo tenemos que iniciar su programa instalador y seguir los pasos de instalación. Esta aplicación trae consigo, tanto el entorno de desarrollo Eclipse, como varias versiones del Servidor de Aplicaciones Web “GlassFish” (en nuestro caso, haremos uso de la versión 2).

A continuación, iniciaremos el servicio del servidor de aplicaciones GlassFish, seleccionando la versión 2, como se muestra en la figura 39:



**Figura 39. Arrancar Servidor GlassFish v2**

Posteriormente, y tras esperar unos instantes a que el servidor se arranque, si no existe ningún paquete de servicios asociados al servidor GlassFish (en el caso de la figura 39 se tiene asociado el paquete de servicios WS\_WikiEAR), se debe generar este por medio del asistente para la creación del Web Service, del modo como se muestra en la figura 40:



**Figura 40. Crear Web Service**

Seguidamente, aparecerá una ventana en la que se podrán seleccionar algunas características durante el proceso de creación y publicación del Web Service. Seleccionamos la opción “Finish” para terminar con el asistente y que se publique el Web Service a través de GlassFish:

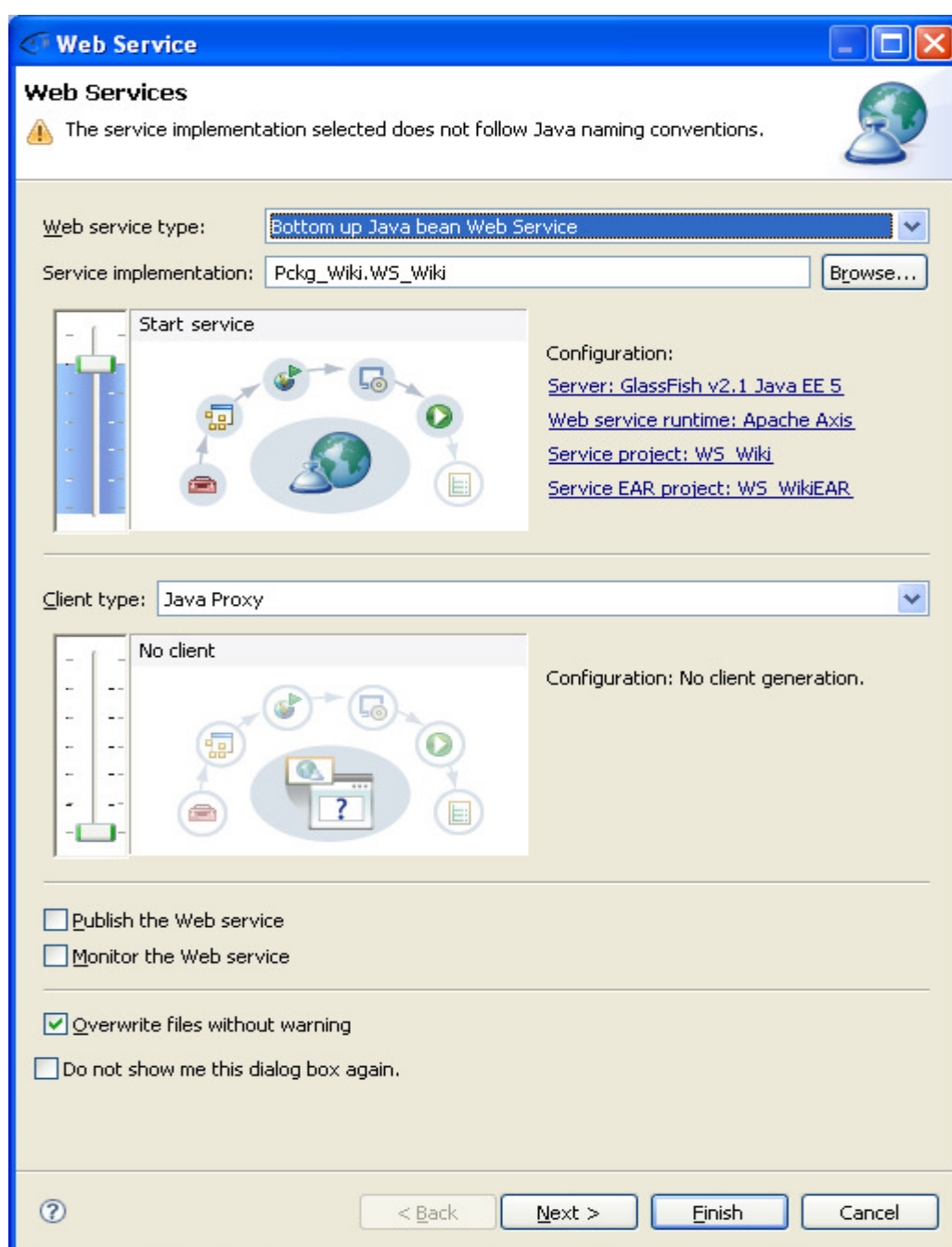


Figura 41. Configurar Web Service

### **1.3.3 Servidor de Aplicaciones Web: GlassFish v2**

Existen dos opciones para instalar el Servidor de Aplicaciones Web GlassFish: la primera opción se explicó en la sección 1.3.2 de este Anexo (Anexo A); la segunda opción consiste en hacerlo de manera individual, realizando la instalación de GlassFish, independientemente del IDE Eclipse, arrancando el servidor manualmente y publicando el Web Service, necesario para el funcionamiento de la herramienta, a través de la aplicación web gratuita “*Sun GlassFish Enterprise Server v2.1*”, que se instala a la vez que el servidor y al que se tiene acceso desde un navegador por medio de el enlace “<http://localhost:4849/>” (ruta por defecto al instalar el servidor de aplicaciones).

En nuestro caso, se optó por la primera opción, por lo que queda fuera de este anexo la explicación del segundo método.

## **1.4 CONFIGURACIONES**

No se requiere ningún tipo de configuración específica para el uso de la herramienta.

## **2 INSTALACIÓN EN EL CLIENTE**

### **2.1 REQUERIMIENTOS MÍNIMOS**

Antes de instalar el componente de la herramienta se debe asegurar de contar con las siguientes características y/o elementos:

- Sistema Operativo: Windows XP o superior
- Microsoft Project 2003
- Acceso a Internet

El equipo servidor debe disponer de, al menos, un microprocesador con velocidad de 2 Gigahercios (GHz) de procesamiento, una memoria RAM de 1 Gigabyte (GB) y un espacio en el disco duro de 2 Megabytes (MB).

### **2.2 INSTALACIÓN DEL HARDWARE**

No se requiere la instalación de ningún componente hardware específico para el uso de la herramienta.

### **2.3 INSTALACIÓN DEL SOFTWARE**

#### **2.3.1 Microsoft Project 2003**

Para instalar MS-Project se debe disponer de la licencia correspondiente y seguir los pasos en el proceso que guía la instalación de la aplicación.

#### **2.3.2 Componente Addin**

La instalación de nuestra herramienta se realizar de manera muy sencilla, por medio de un programa instalador que permite seleccionar la ruta de destino de los archivos necesarios para su correcto funcionamiento.

El registro del Addin en el sistema, y su correspondiente asociación con la aplicación MS-Project, se realiza de manera automática y transparente al usuario durante el proceso de instalación.

## **2.4 CONFIGURACIONES**

No se requiere ningún tipo de configuración específica para el uso de la herramienta.



## **ANEXO B. MANUAL DE USUARIO**

---



# 1 INICIO DEL USO DE LA HERRAMIENTA

Una vez iniciada la aplicación Microsoft Project, podemos elegir entre dos opciones para realizar la llamada a nuestra herramienta y comenzar a hacer uso de ella:

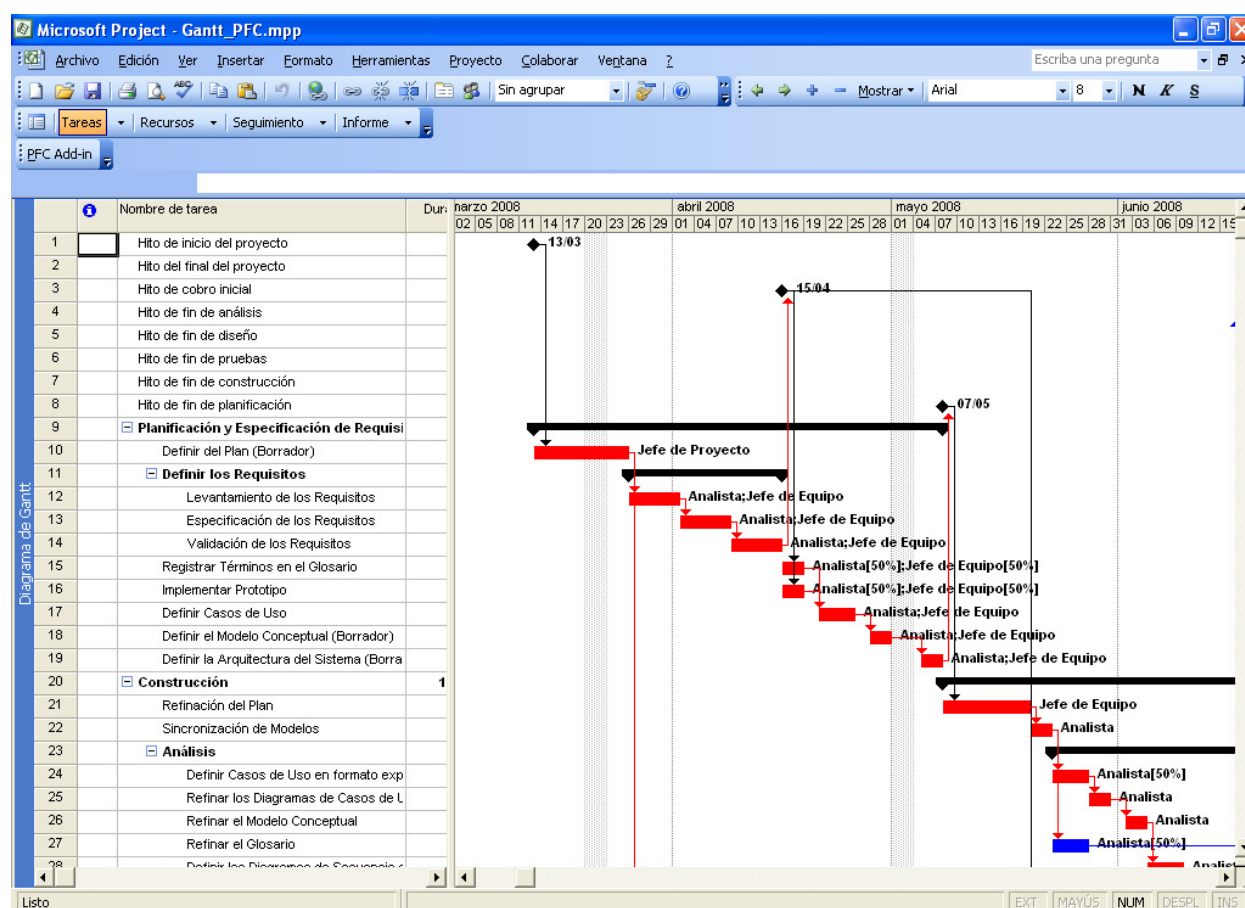


Figura 42. Inicio de Microsoft Project

- a) Mediante un acceso, llamado *PFC-Addin*, existente en la “Barra de Herramientas”:

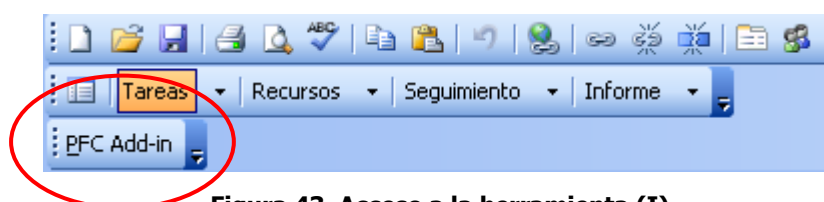
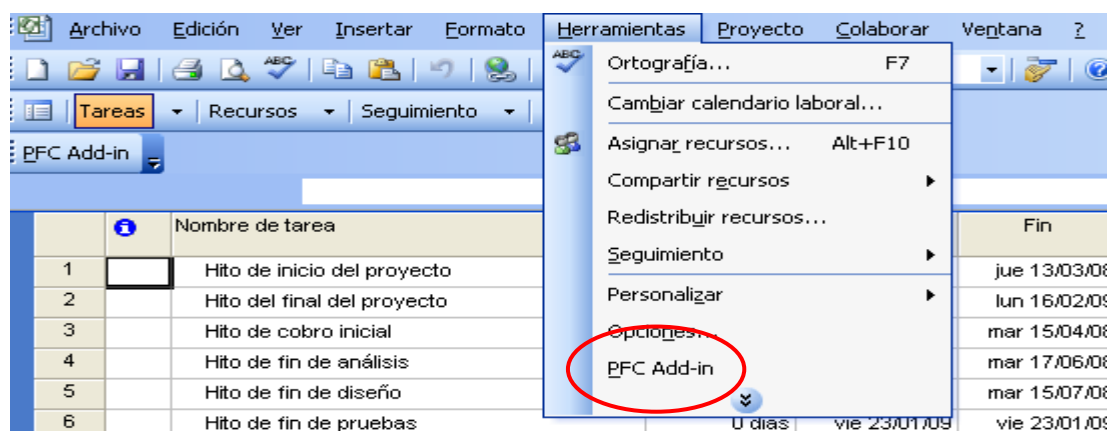


Figura 43. Acceso a la herramienta (I)

b) Mediante el acceso existente en la “Barra de Menú” (Herramientas/PFC-Addin)



**Figura 44. Acceso a la herramienta (II)**

## 2 IDENTIFICACIÓN Y PRUEBA DEL FICHERO WSDL

Una vez realizada la llamada a nuestra herramienta, se mostrará una ventana de conexión con la base de datos y de testeo de la ruta del fichero público WSDL, tal y como se muestra en la figura 45:

**Conexión con el Servidor**

**PARÁMETROS DE CONEXIÓN:**

(\*) Servidor WSDL:

(\*) Usuario:

(\*) Contraseña:

(\*) Campo Obligatorio

**Figura 45. Conexión y Prueba WSDL**

Parámetros requeridos en esta pantalla son los siguientes:

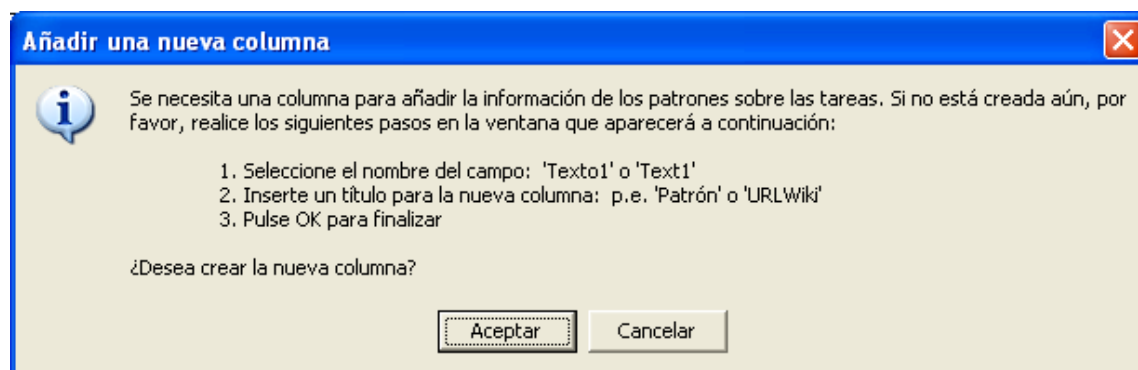
- Servidor WSDL: ruta del fichero WSDL, proveedor de los servicios del Web Service.
- Usuario y Contraseña: realiza la conexión con la BBDD y verifica si son correctos.

Tanto en la función de testeo de la ruta del servidor WSDL, como en la de autenticación del usuario y contraseña introducida, en el caso de existir algún problema, bien sea de comunicación, bien sea de datos y/o rutas incorrectas, se mostrará un mensaje informando de la situación particular.

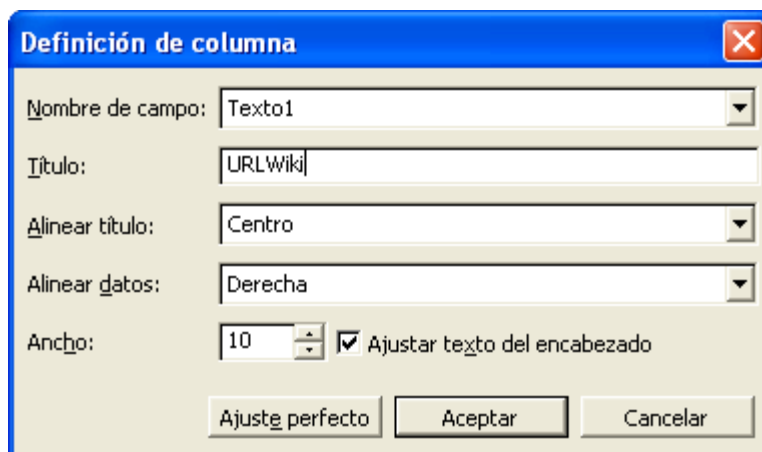
Tras realizar la validación correcta de los datos, se muestra la opción de crear la nueva columna donde se almacenarán los enlaces de los patrones, seleccionados en los siguientes pasos de la herramienta, tal y como se muestra, en secciones más avanzadas, dentro del manual de uso.

Así pues, se mostrará una ventana informativa mostrándonos la opción de crear la columna en ese instante (esta se puede realizar en otros momentos, durante el uso de la herramienta), por medio de un asistente/guía.

Las figuras 45 y 46 muestran la ventana informativa y el asistente/guía para la creación del campo dentro del MS-Project:

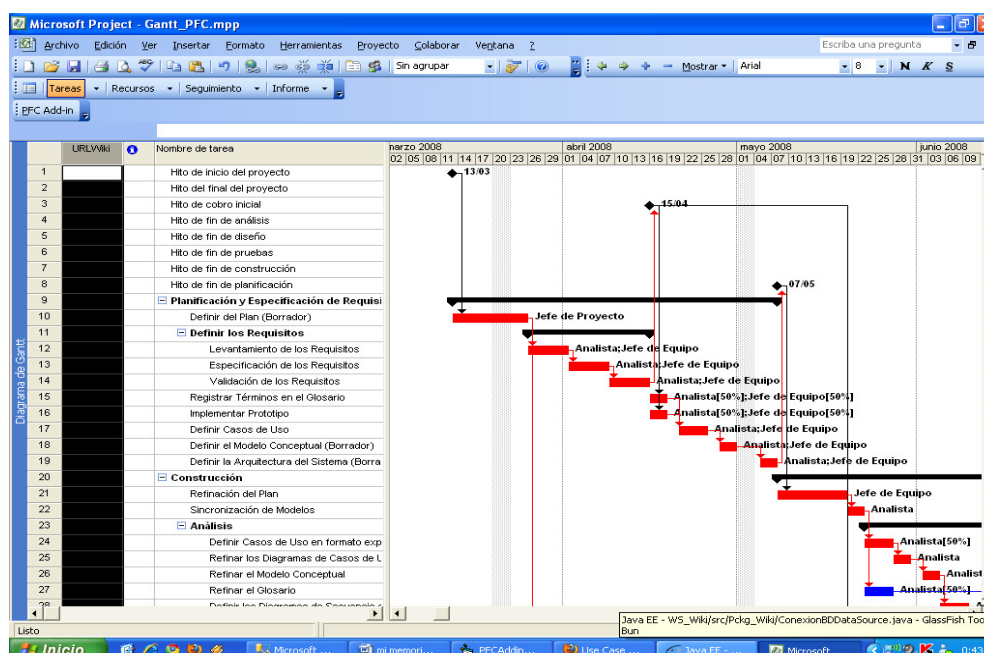


**Figura 46. Asistente para la creación de la Columna Destino (I)**



**IMPORTANTE:** Para añadir la columna que contendrá la ruta del Patrón de Producto dentro del cronograma, debe seleccionarse el valor “*Texto1*” del campo “*Nombre de Campo*”.

Finalmente, la aplicación MS-Project queda como se muestra en la figura 48:



### 3 SELECCIÓN DE PARÁMETROS PARA LA BÚSQUEDA DE PATRONES DE PRODUCTO

Para realizar la consultar de Patrones de Producto será necesario seleccionar varios parámetros:

- Contexto
- Restricción Genérica (o de Organización)
- Restricción Específica (o de Proyecto)
- Tarea Origen

En la figura 49 se puede observar la pantalla de donde se deben seleccionar los tres primeros elementos para la búsqueda. Inicialmente se muestra un listado de todos los Contextos y Restricciones Genéricas existentes en la base de datos. El listado de Restricciones Específicas viene determinado por la Restricción Genérica seleccionada previamente.

La imagen muestra una ventana de software titulada "Selección del Contexto y Restricciones".

- PASO 1.1: Seleccione uno de los Contextos**
  - Un campo etiquetado "(\*) Contexto" contiene "Context 01".
  - Debajo, una lista desplegable muestra "Context 01", "Context 02" y "Context 03".
- PASO 1.2: Seleccione**
  - Un campo etiquetado "(\*) Restricciones" está vacío.
  - Debajo, hay dos paneles de selección:
    - A) De Organización (GENÉRICA)**: Muestra una lista con "GenericForce 01", "GenericForce 02", "GenericForce 03" y "GenericForce 04".
    - B) De Proyecto (ESPECÍFICA)**: Muestra una lista con "SpecificForce 01", "SpecificForce 02" y "SpecificForce 03".

En la parte inferior derecha, hay una leyenda "(\*) Campo Obligatorio". En la parte inferior, hay dos botones: "Aceptar" y "Salir".

Figura 49. Selección del Contexto y las Restricciones

Solo se necesita seleccionar la Tarea Origen para comenzar la búsqueda de los Patrones de Producto que coincidan con los parámetros de selección introducidos.

**Selección de Parámetros de la Búsqueda**

**PASO 1: Seleccione las Restricciones**

Contexto	Context 01
Restricción Genérica	GenericForce 01
Restricción Específica	SpecificForce 01

Modificar

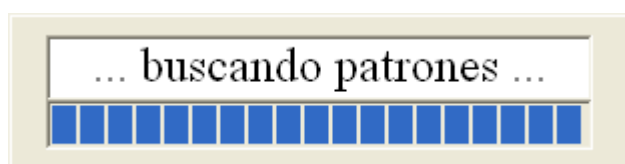
**PASO 2: Seleccione las Tarea Origen de la Consulta**

- Hito de inicio del proyecto
- Hito del final del proyecto
- Hito de cobro inicial
- Hito de fin de análisis
- Hito de fin de diseño
- Hito de fin de pruebas
- Hito de fin de construcción
- Hito de fin de planificación
- Planificación y Especificación de Requisitos
- Definir el Plan (Borrador)
- Definir los Requisitos
- Levantamiento de los Requisitos
- Especificación de los Requisitos
- Validación de los Requisitos
- Registrar Términos en el Glosario
- Implementar Prototipo
- Definir Casos de Uso**
- Definir el Modelo Conceptual (Borrador)
- Definir la Arquitectura del Sistema (Borrador)
- Construcción
- Refinación del Plan
- Sincronización de Modelos
- Análisis
- Definir Casos de Uso en formato expandido

Buscar Salir

**Figura 50. Selección de los Parámetros de Búsqueda**

Mientras dura el proceso de búsqueda, se muestra un elemento visual para que el Jefe de Proyecto tenga conocimiento de que la operación se está llevando a cabo:



**Figura 51. Mensaje de estado (I)**

## 4 SELECCIÓN Y ASIGNACIÓN DE PATRONES DE PRODUCTO

Una vez realizada la operación de búsqueda se obtienen los Patrones de Producto relacionados, tal y como se muestra, a modo de ejemplo, en la figura 52:

**Búsqueda y Asignación de Patrones**

**PASO 3: Creación de la columna destino [OPCIONAL]**

Cree la columna destino de la actualización

**PASO 4: Selección de patrón**

**Patrones**

- Expanded Use Case Diagram
- High-Level Use Case Diagram
- Use Case Diagram

**Descripción**

- The Use Case Diagram have to satisfy the requirements.
- It shows the way the system is going to be used.
- Identify the interactions between the system and its actors.

**URLWiki**

[http://productpatterns.sel.inf.uc3m.es/Product\\_Patte](http://productpatterns.sel.inf.uc3m.es/Product_Patte)

**PASO 5: Ámbito de actualización**

Seleccione la/s tarea/s a que desea actualizar

**PASO 6: Almacenar proyecto en el sistema**

Guarde en el sistema los cambios del proyecto

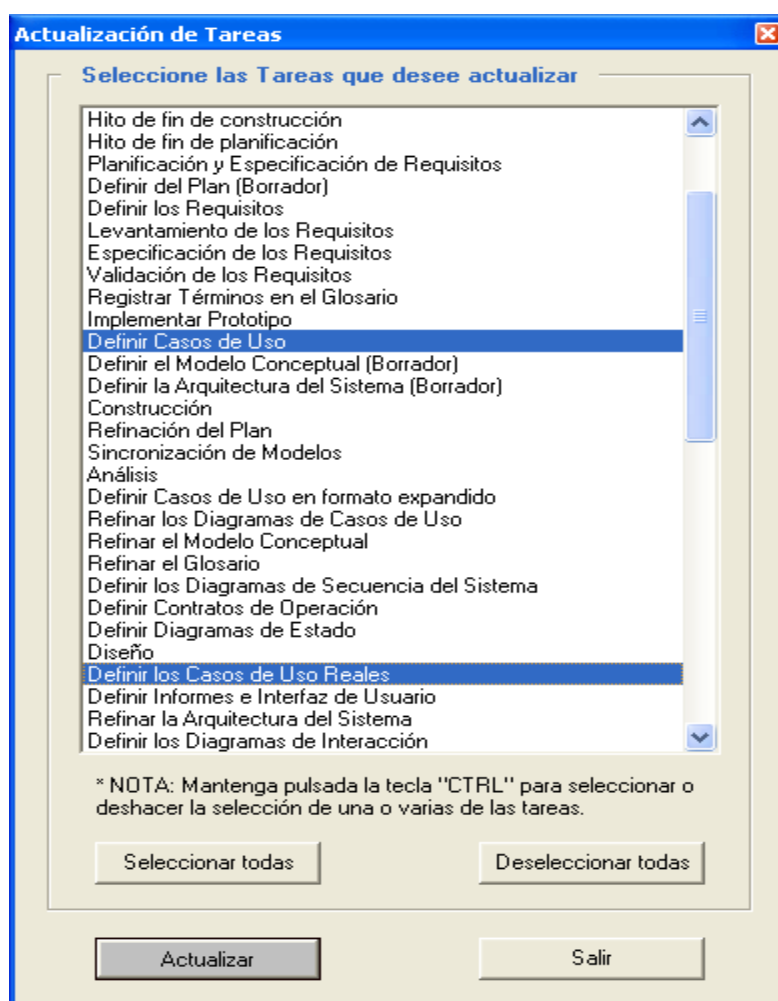
Volver Finalizar

**Figura 52. Selección de Patrones**

Desde esta ventana se puede visualizar la descripción de los Patrones o acceder a su enlace URL (contenido en una Wiki), el cuál contiene toda la información con un mayor grado de detalle (tareas, sub-tareas, todos los contextos relacionados, roles, problemas, soluciones, recursos, etc....).

En el momento en el que el Jefe de Proyecto tiene seleccionado el Patrón que mejor se adapta a sus necesidades, nos encontramos en condiciones de realizar la selección de las tareas que serán objeto de la asignación con la URL de dicho Patrón. Para ello, el Jefe de Proyecto debe seleccionar el botón del “Paso 5: Ámbito de la actualización”.

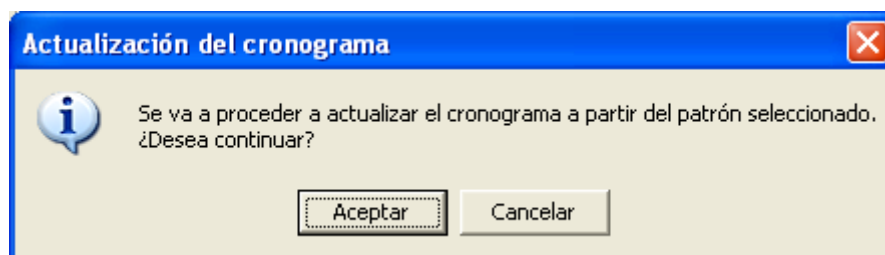
La operación de selección de tareas destino se realiza a través de la siguiente ventana:



**Figura 53. Selección de las Tareas Destino de la actualización**



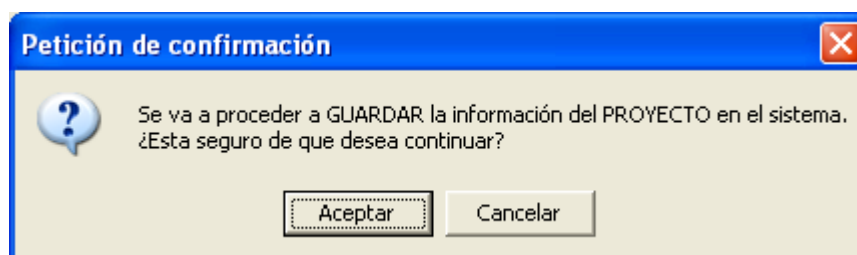
Antes de dar por finalizada la actualización, la herramienta pide confirmación al Jefe de Proyecto para realizar la actualización del campo asociado a las tareas, dentro de la planificación:



**Figura 54. Mensaje de confirmación (I)**

## 5 GUARDAR HISTÓRICO DE LA PLANIFICACIÓN

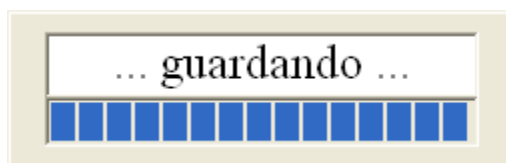
Para guardar las características de la planificación modificada con los Patrones de Producto, el Jefe de Proyecto debe seleccionar el botón del “Paso 6: Almacenar proyecto en el sistema”, tras lo cual aparecerá un mensaje solicitando la confirmación de la operación, tal y como se muestra en la figura 55:



**Figura 55. Mensaje de confirmación (II)**

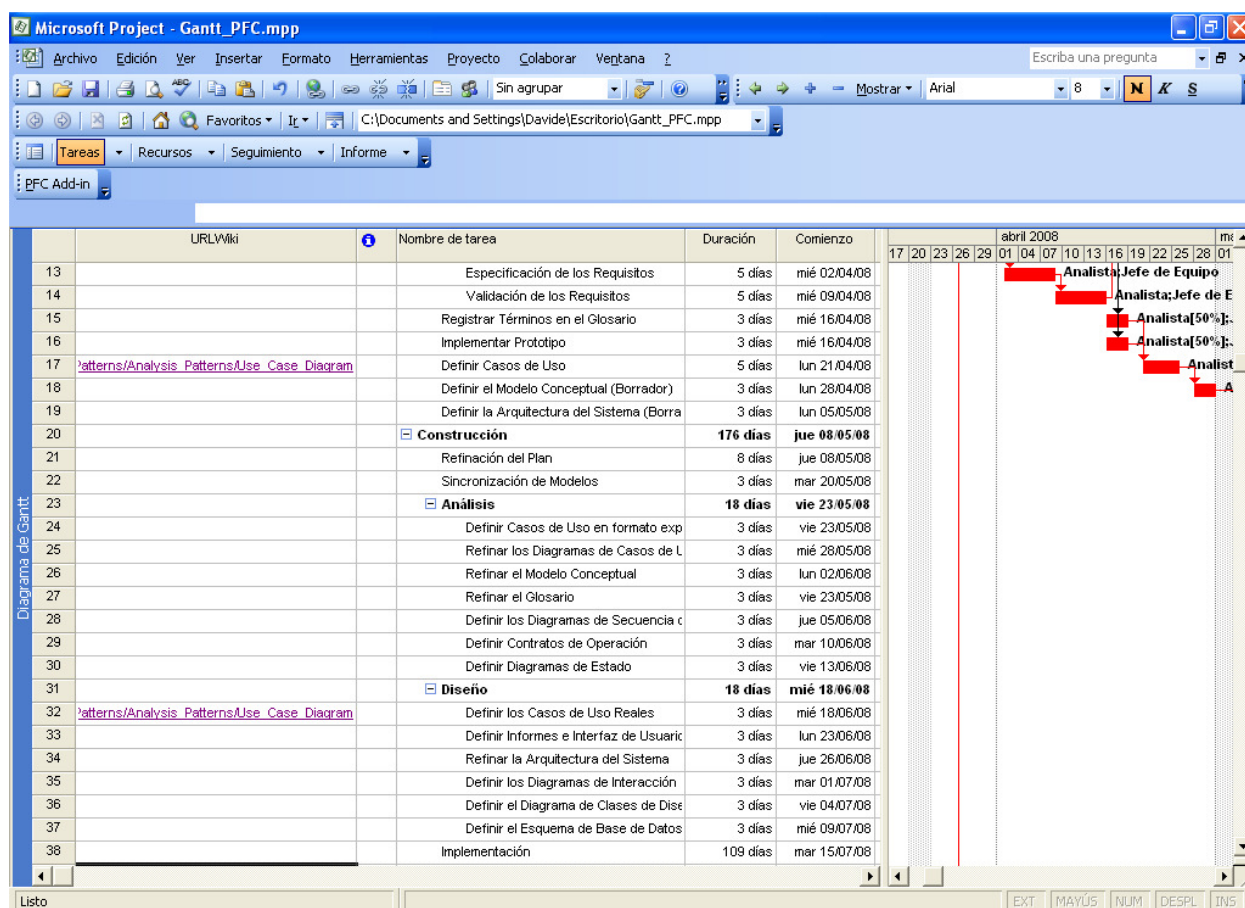
A partir de este momento, se registra el histórico del proyecto en la base de datos.

Mientras dura el proceso de almacenamiento, se muestra un elemento visual para que el Jefe de Proyecto tenga conocimiento de que la operación se está llevando a cabo:



**Figura 56. Mensaje de estado (II)**

La figura 57 muestra como queda la actualización de la planificación, sobre MS-Project, tras terminar el uso de nuestra herramienta:



**Figura 57. Calendario con columna URLWiki rellena**

